

**НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ
імені ІГОРЯ СІКОРСЬКОГО»**

Факультет прикладної математики

Кафедра програмного забезпечення комп'ютерних систем

«На правах рукопису»
УДК 004.9

«До захисту допущено»

Науковий керівник кафедри

_____ Іван ДИЧКА

«__» _____ 2020 р.

Магістерська дисертація

на здобуття ступеня магістра

за освітньо-професійною програмою

**«Інженерія програмного забезпечення комп'ютерних та
інформаційно-пошукових систем»**

зі спеціальності 121 Інженерія програмного забезпечення

**на тему: «Програмний метод формування рекомендацій рекламних
оголошень на основі аналізу вподобань користувача»**

Виконав:

студент II курсу, групи КП-91мп
Дяченко Дмитро Олександрович _____

Керівник:

Старший викладач кафедри ПЗКС, к.т.н.,
Рибачок Наталія Антонівна _____

Консультант з нормоконтролю:

Доцент кафедри ПЗКС, к.т.н., доцент,
Онай Микола Володимирович _____

Рецензент:

Доцент кафедри електронної інженерії, к.т.н., доцент,
Вунтесмері Юрій Володимирович _____

Засвідчую, що у цій магістерській
дисертації немає запозичень з праць
інших авторів без відповідних
посилань.

Студент _____

Київ – 2020 року

Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського»
Факультет прикладної математики

Кафедра програмного забезпечення комп'ютерних систем

Рівень вищої освіти – другий (магістерський)

Спеціальність – 121 «Інженерія програмного забезпечення»

Освітньо-професійна програма «Інженерія програмного забезпечення комп'ютерних та інформаційно-пошукових систем»

ЗАТВЕРДЖУЮ

Науковий керівник кафедри

_____ Іван ДИЧКА

«__» _____ 2019 р.

ЗАВДАННЯ
на магістерську дисертацію студенту

Дяченко Дмитру Олександровичу

1. Тема дисертації «Програмний метод формування рекомендацій рекламних оголошень, на основі аналізу вподобань користувача», науковий керівник дисертації Рибачок Наталія Антонівна, к.т.н., доцент, затверджені наказом по університету від «12» листопада 2020 р. №3298-С
2. Термін подання студентом дисертації «14» грудня 2020 р.
3. Об'єкт дослідження: процес формування особистих рекламних рекомендацій на основі вподобань користувачів.
4. Предмет дослідження: методи формування персональних рекомендацій.
5. Перелік завдань, які потрібно розробити:
 - провести аналіз методів проектування рекомендаційних систем;
 - провести аналіз методів формування рекомендацій рекламних оголошень;
 - розробити модифікований гібридний метод формування рекомендацій рекламних оголошень на основі аналізу вподобань користувача;
 - обрати критерії оцінки методу та провести тестування роботи методу;
 - порівняти ефективність запропонованого методу з іншими;
 - оцінити якість роботи методу в залежності від різних параметрів системи;
 - провести аналіз отриманих результатів.
6. Орієнтовний перелік графічного (ілюстративного) матеріалу:
 - схема алгоритму роботи модифікованого методу;
 - ER-діаграма бази даних;
 - діаграма класів;
 - діаграма послідовності;
 - діаграма варіантів використання системи;

7. Орієнтовний перелік публікацій:

- Тези доповіді “ Програмний метод формування рекомендацій рекламних оголошень на основі аналізу вподобань користувача”

8. Консультанти розділів дисертації

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
Нормоконтроль	Онай М.В., доцент кафедри ПЗКС		

9. Дата видачі завдання «11» жовтня 2019 р.

Календарний план

№ з/п	Назва етапів виконання магістерської дисертації	Термін виконання етапів магістерської дисертації	Примітка
1.	Грунтовне ознайомлення з предметною галуззю	17.10.2019	
2.	Визначення структури магістерської дисертації; вивчення літератури, пошук додаткової літератури, патентний пошук	04.12.2019	
3.	Робота над першим розділом магістерської дисертації; проведення наукового дослідження	15.02.2020	
4.	Проведення наукового дослідження; робота над другим розділом магістерської дисертації; розроблення програмного забезпечення	05.04.2020	
5.	Проведення наукового дослідження; робота над статтею за результатами наукового дослідження	15.05.2020	
6.	Проведення наукового дослідження; робота над третім розділом магістерської дисертації	15.06.2020	
7.	Завершення роботи над основною частиною магістерської дисертації; підготовка ілюстративного матеріалу; підготовка матеріалів доповіді на конференції ПМК-2020	05.11.2020	
8.	Оформлення текстової і графічної частини магістерської дисертації	04.12.2020	

Студент

Дмитро ДЯЧЕНКО

Науковий керівник

Наталія РИБАЧОК

РЕФЕРАТ

Актуальність. Наразі людство вступило в інформаційну епоху, де найбільш цінним ресурсом є інформація. Але об'єми інформації зростають набагато швидше, ніж обчислювальні можливості для обробки цієї інформації. Кількість інформації в мережі настільки велика, що людині дуже важко отримати те, що її дійсно цікавить.

У зв'язку з цим, актуальною є розробка спеціальної рекомендаційної системи, що пропонуватиме рекламні оголошення, які відповідатимуть інтересам користувача. Такі персональні рекомендації формуються на основі аналізу вподобань поточного користувача та інших користувачів.

Об'єктом дослідження є процес формування особистих рекламних рекомендацій на основі вподобань користувачів.

Предметом дослідження є методи формування персональних рекомендацій.

Мета роботи: підвищення ефективності роботи рекламної системи за рахунок розробки програмного методу формування рекомендацій на основі аналізу вподобань користувача.

Методи дослідження застосовані у даній роботі базуються на методах інформаційного пошуку.

Наукова новизна роботи полягає у застосуванні гібридного підходу до фільтрації даних для формування персональних рекомендацій. Розроблено підхід, який поєднує в собі фільтрацію даних на основі вмісту з колаборативною фільтрацією.

Практична цінність отриманих в роботі результатів полягає в тому, що запропонований метод дозволить отримувати користувачам персоналізовані рекомендації рекламних оголошень, а рекламним платформам підвищити ефективність своїх оголошень.

Апробація роботи. Основні положення і результати роботи були представлені та обговорювались на XIII науковій конференції магістрантів

та аспірантів «Прикладна математика та комп'ютинг» ПМК-2020 (Київ, 18-20 листопада 2020 р.).

Структура та обсяг роботи. Магістерська дисертація складається з вступу, п'яти розділів, висновків та додатків.

У вступі надано загальну характеристику роботи, виконано оцінку сучасного стану проблеми, обґрунтовано актуальність напрямку досліджень, сформульовано мету і задачі досліджень.

У першому розділі проведено аналіз основних методів формування рекомендацій, які можуть бути використані при розробці модифікованого гібридного методу.

У другому розділі описано розроблений модифікований метод формування рекомендацій на основі аналізу вподобань користувачів, та наведено алгоритм роботи методу.

У третьому розділі описано засоби розробки програмного забезпечення, архітектуру програмного забезпечення, структуру бази даних, діаграму класів, діаграму послідовності роботи системи та діаграму компонентів системи.

У четвертому розділі описано набори даних, які використовувалися для проведення експериментів. Визначено характеристики релевантності, які використовувалися для оцінки ефективності роботи існуючих та розробленого методів формування рекомендацій. Наведено результати проведених експериментів.

У п'ятому розділі проведено опис бізнес-моделі, огляд існуючих аналогів систем, які використовують методи формування рекомендацій, ідеї стартап-проекту, а також проведено аналіз ринкових можливостей запуску стартап-проекту.

У висновках проаналізовано результати, отримані під час виконання роботи.

У додатках наведено схему алгоритму роботи модифікованого методу формування рекомендацій, ER-діаграму бази даних, діаграму класів,

діаграму послідовності, діаграму варіантів використання системи та копію презентації.

Робота виконана на 124 аркушах, містить 3 додатків та посилання на список використаних літературних джерел з 37 найменувань. У роботі наведено 12 рисунків та 16 таблиць.

Ключові слова: рекомендаційна система, метод, модель, рекомендації, колаборативна фільтрація, фільтрація на основі вмісту, демографічна фільтрація, пошук, рейтинг, схожість.

ABSTRACT

Theme urgency. Nowadays, humanity has entered the information age, where the most valuable resource is information. But the amount of information is growing much faster than the computing power to process that information. The amount of information on the network is so large that it is very difficult for a person to get what he is really interested in.

Therefore, it is important to develop a special recommendation system that will offer advertisements that will meet the interests of the user. Such personal recommendations are formed on the basis of the analysis of preferences of the current user and other users.

Object of research is the process of forming personal advertising recommendations based on user preferences.

Subject of research are the methods of forming personal recommendations.

Research objective: improving the efficiency of the advertising system by developing a software method of forming recommendations based on the analysis of user preferences.

Research methods applied in these work are based on information retrieval methods.

Scientific novelty of the work is the application of a hybrid approach to data filtering for the formation of personal recommendations. An approach has been developed that combines content-based data filtering with collaborative filtering.

Practical value of the results obtained in this work is that the proposed method will allow users to receive personalized recommendations for advertisements, and advertising platforms to increase the effectiveness of their ads.

Approbation. The main provisions and results of the work were presented and discussed at the XIII scientific conference of undergraduates and graduate students "Applied Mathematics and Computing" PMK-2020 (Kyiv, November 18-20, 2020).

Structure and content of the thesis. The master's thesis consists of an introduction, five chapters, conclusions and appendices.

The introduction provides a general description of the work, performed an assessment of the current state of the problem, substantiated the relevance of the direction research, formulated the purpose and objectives of research.

The first section analyzes the main methods of forming recommendations that can be used in the development of a modified hybrid method.

The second section describes the developed modified method of forming recommendations based on the analysis of user preferences, and presents the algorithm of the method.

The third section describes software development tools, software architecture, database structure, class diagram, system sequence diagram, and system component diagram.

The fourth section describes the data sets used for the experiments. The characteristics of relevance that were used to assess the effectiveness of existing and developed methods of forming recommendations are identified. The results of the conducted experiments are given.

The fifth section describes the business model, an overview of existing analogues of systems that use methods of forming recommendations, ideas of a startup project, as well as an analysis of market opportunities to start a startup project.

The conclusions analyze the results obtained during the work.

The appendices provide a diagram of the algorithm of the modified method of forming recommendations, ER-diagram of the database, class diagram, sequence diagram, diagram of options for using the system and a copy of the presentation.

The work is done on 124 sheets, contains 3 appendices and links to a list of used literature sources from 37 titles. The paper presents 12 figures and 16 tables.

Key words: recommendation system, method, model, recommendations, collaborative filtering, content-based filtering, demographic filtering, search, rating, similarity.

ЗМІСТ

СПИСОК ТЕРМІНІВ, СКОРОЧЕНЬ ТА ПОЗНАЧЕНЬ	3
ВСТУП.....	6
1. АНАЛІЗ ІСНУЮЧИХ МЕТОДІВ ФОРМУВАННЯ РЕКОМЕНДАЦІЙ...	8
1.1. Опис задачі формування рекомендацій.....	8
1.2. Математична постановка задачі знаходження найбільш релевантного об'єкта.....	9
1.3. Огляд методів надання рекомендацій	10
1.4. Висновки до розділу.....	20
2. МОДИФІКОВАНИЙ МЕТОД ФОРМУВАННЯ РЕКОМЕНДАЦІЙ	21
2.1. Модифікований метод формування рекомендацій	21
2.2. Опис модифікованого методу	24
2.3. Висновки до розділу.....	27
3. АНАЛІЗ ПРОГРАМНОЇ РЕАЛІЗАЦІЇ	28
3.1. Засоби розробки програмного забезпечення	28
3.2. Структура бази даних.....	30
3.3. Архітектура розробленого програмного забезпечення	42
3.4. Опис технічного забезпечення.....	46
3.5. Висновки до розділу.....	46
4. АНАЛІЗ ОТРИМАНИХ РЕЗУЛЬТАТІВ.....	48
4.1. Тестування програмного продукту.....	48
4.2. Оцінка роботи методу	48
4.3. Висновки до розділу.....	55
5. ПОБУДОВА БІЗНЕС-МОДЕЛІ	57

5.1. Опис бізнес-процесів.....	57
5.2 Опис постановки задачі	63
5.3 Огляд аналогів системи	64
5.4 Опис ідеї стартап-проекту	66
5.5 Аналіз ринкових можливостей запуску стартап-проекту	66
5.6 Висновки до розділу.....	68
ВИСНОВКИ.....	70
СПИСОК ЛІТЕРАТУРНИХ ДЖЕРЕЛ	72

СПИСОК ТЕРМІНІВ, СКОРОЧЕНЬ ТА ПОЗНАЧЕНЬ

Архітектура ПЗ (англ. software architecture) – структура, що включає в себе елементи програмного забезпечення, взаємозв'язки між ними та властивості цих елементів та взаємозв'язків між ними [1].

БД (англ. database) – база даних, організована колекція даних та зв'язків між ними, що дозволяє зберігати їх та мати доступ до них у електронному форматі [2].

ПЗ (англ. software) – програмне забезпечення, програма або сукупність програм, що використовуються для управління комп'ютером [3].

Фреймворк (англ. framework) – це каркас або шаблон, що використовується для полегшення та прискорення розроблення програмного забезпечення та об'єднання різних модулів всередині проекту. Він визначає структуру програми [4].

СКБД (англ. Database Management System) – система керування базами даних, представляє собою організовану сукупність програмних та лінгвістичних даних (схеми, таблиці, запити, звіти, перегляди, тощо). Це програма, що дозволяє взаємодіяти із базами даних як абстрактним об'єктом [5].

СВ – колаборативна фільтрація, метод, який використовується деякими рекомендаційними системами, процес фільтрації інформації або зразків за допомогою методів за участю співробітництва між декількома агентами, точками зору, джерелами даних і т.д. [6].

CF – фільтрація на основі контенту, метод, який використовується деякими рекомендаційними системами, процес фільтрації інформації за допомогою даних про елементи системи.

CSS (англ. Cascading Style Sheets) – каскадні таблиці стилів, формальна мова, що застосовується для опису зовнішнього вигляду

документа (веб-сторінки), написаний із використанням мов розмітки. Може застосовуватися до XML-документу [7].

DF – демографічна фільтрація, метод, який використовується в рекомендаційних системах, що використовує демографічні дані про користувачів для фільтрації інформації за участю співробітництва між декількома агентами, точками зору, джерелами даних і т.д.

ER-модель (англ. Entity-Relationship model) – модель даних, що дозволяє описати концептуальні схеми предметної області [8].

HTML (англ. HyperText Markup Language) – мова гіпертекстової розмітки, стандартна мова розмітки документів, призначених для відображення у веб-браузері. Разом із ним використовуються такі технології, як каскадні таблиці стилів та мови сценаріїв (JavaScript) [9].

HTTP – HyperText Transfer Protocol, це протокол прикладного рівня, де обмін повідомленнями йде за звичайною схемою запит-відповідь.

MongoDB – документо-орієнтована система керування базами даних (СКБД) з відкритим вихідним кодом, яка не потребує опису схеми таблиць. MongoDB займає нішу між швидкими і масштабованими системами, що оперують даними у форматі ключ/значення, і реляційними СКБД, функціональними і зручними у формуванні запитів [10].

Node.js – платформа з відкритим кодом для виконання високопродуктивних мережевих застосунків, написаних мовою JavaScript [11].

ORM (англ. Object-relational mapping) – об'єктно-реляційна проекція, технологія програмування, що здійснює перетворення даних між системами несумісного типу за допомогою об'єктно-орієнтованих мов програмування [12].

IDEF0 (англ. Function Modeling) – методологія функціонального моделювання і графічного описання процесів, призначена для формалізації і опису бізнес-процесів. В IDEF0 розглядаються логічні зв'язки між роботами, а не послідовність їх виконання в часі [13].

TF-IDF міра (англ. TF – term frequency, IDF – inverse document frequency) – статистичний показник, що використовується для оцінки важливості слів у контексті документа, що є частиною колекції документів чи корпусу. Вага (значимість) слова пропорційна кількості вживань цього слова у документі, і обернено пропорційна частоті вживання слова у інших документах колекції [14].

JavaScript – це об'єктно-орієнтована, прототипна та динамічна мова програмування, реалізація стандарту ECMAScript [15].

JSON (англ. JavaScript Object Notation) – запис об'єктів JavaScript, це текстовий формат обміну даними між комп'ютерами. JSON базується на тексті, може бути прочитаним людиною. Формат дає змогу описувати об'єкти та інші структури даних.

ВСТУП

На сьогоднішній день, людство живе в інформаційну епоху, тому найбільшим цінним ресурсом є інформація. Інформація відіграє одну з важливіших ролей у процесі формування сучасного суспільства. Після того, як було винайдено Інтернет, основним сховищем інформації стала Всесвітнє Павутиння. Але з плином часу, об'єми інформації в мережі Інтернет почали зростати набагато швидше ніж обчислювальні можливості, для обробки цієї інформації. Зараз, кількість інформації в Інтернеті настільки велика, що людині дуже важко отримати те, що дійсно її цікавить.

Сьогодні кількість інтернет-ресурсів неможливо підрахувати, кожную хвилину їх стає все більше і більше. Кожен такий ресурс представляє собою малий або великий обсяг даних. Усю цю інформацію потрібно обробляти. За допомогою цього можливо підвищити ефективність системи. Цей процес відбувається шляхом аналізу поведінок та вподобань користувачів електронних ресурсів, для того щоб спрогнозувати їх побажання у майбутньому та надати їм персоналізовані рекомендації. Такий тип системи називається – рекомендаційна система.

Завданням рекомендаційних системи є процес формування рекомендацій товарів, послуг або іншого, для користувача, на основі переваг самого користувача, або на основі переваг інших користувачів, схожих на користувача за купленими товарами, послугами. Так як рекомендаційні системи різних типів є поширеними, тема є актуальною.

Актуальною є розробка спеціальної рекомендаційної системи, яка буде пропонувати користувачу деякі елементи, які будуть вважатися доцільними та цікавими для користувача. Система буде формувати персональні рекомендації на основі аналізу вподобань користувачів. Це дозволить користувачам отримувати персоналізовані рекомендації, які будуть відповідати їх інтересам.

Отже, можна зробити висновок, що тема досліджень в області рекомендаційних систем є досить популярною і актуальною. Станом на сьогодні рекомендаційні системи стрімко розвиваються, а саме якість їх роботи постійно збільшується. Тому в якості дослідження є необхідним провести аналіз найкращих моделей, методів та практик для підвищення ефективності надання персональних рекомендацій користувачам.

1. АНАЛІЗ ІСНУЮЧИХ МЕТОДІВ ФОРМУВАННЯ РЕКОМЕНДАЦІЙ

1.1. Опис задачі формування рекомендацій

Для реалізації правильної роботи рекомендаційної системи застосовують комплексний підхід, який включає в себе використання принципів роботи та інформації з багатьох інших сфер знань. Наприклад: прогнозування, машинне навчання, інформаційний пошук, когнітивістика, та інші. Також варто відзначити, що рекомендаційні системи зазнали великого впливу з боку бізнесу, так як результати маркетингових досліджень є таким же цінним, як інформаційний аналіз даних. Але рекомендаційні системи завжди використовувалися для отримання рейтингу певних об'єктів для користувача. Використання рекомендаційних систем можливе в якості нової інформації, тобто передбачення вподобань користувача щодо нових елементів, так і в якості використання для корегування роботи системи, тобто порівняння реальної оцінки від користувача з прогнозованою оцінкою рекомендаційної системи.

Для правильної роботи рекомендаційної системи, потрібні початкові дані, на яких будуть базуватися рекомендації. Перш за все це дані про користувачів та елементи всередині системи, а саме рекламні оголошення. Повний цикл роботи рекомендаційної системи починається з отримання даних про рекламні оголошення та користувачів, та закінчується наданням персональних рекомендацій користувачу рекламних оголошень на основі аналізу його вподобань. Вибір методу формування рекомендацій також залежить від інформації, якою буде оперувати система.

Отже, рекомендаційна система – це система, яка націлена на передбачення вподобань користувачів щодо елементів, що є основним контентом системи, у нашому випадку рекламних оголошень, за рахунок наявної інформації про користувача та рекламні оголошення всередині системи [17].

Змістовну постановку задачі формування рекомендацій можна представити у вигляді IDEF0 діаграми (рис. 1).

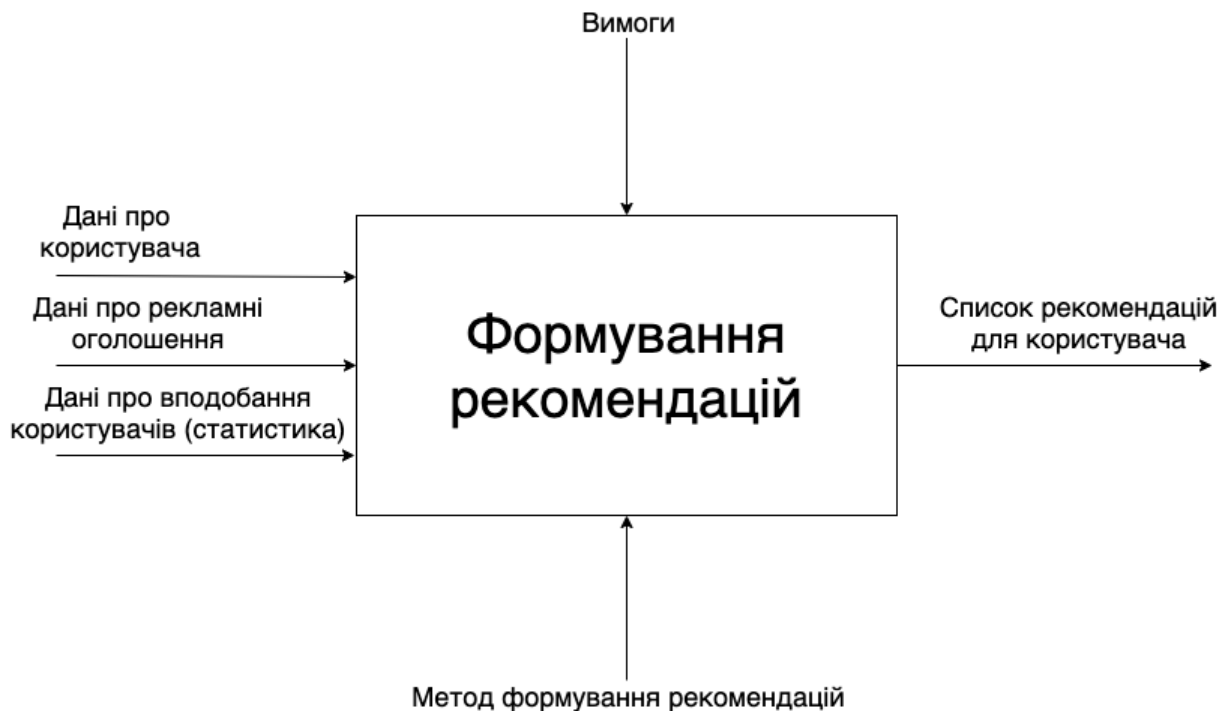


Рис. 1. IDEF0 діаграма роботи системи

1.2. Математична постановка задачі знаходження найбільш релевантного об'єкта

Математичну задачу знаходження найбільш релевантного об'єкта можна описати наступним чином:

$$\forall u \in U, s'_u = \arg \max_{s \in S} h(u, s), \quad (1)$$

де U – це множина користувачів, u – активний (вибраний) користувач, S – множина елементів системи, а саме рекламних оголошень, які потенційно можуть бути рекомендовані користувачу, s – вибраний елемент системи (у нашому випадку рекламне оголошення), h – функція, яка показує ступінь зіставлення деякого елементу s з деяким користувачем u .

Отже, задача зводиться до знаходження такого елементу $s' \in S$, для якого ступінь зіставлення з користувачем $u \in U$ є максимальним, якщо

потрібно рекомендувати один елемент, або до знаходження масиву елементів $[s] \in \mathcal{S}$, якщо потрібно рекомендувати декілька елементів.

Також, варто зазначити, що на точність прогнозованих рекомендацій суттєво впливає не тільки інформація про рекламні оголошення та користувача, а і вибір функції h .

1.3. Огляд методів надання рекомендацій

Для розв'язання поставленої задачі існують різні методи. Можемо виділити основні з них:

- фільтрація на основі вмісту (item-based);
- колаборативна фільтрація (user-based);
- гібридний підхід.

Розглянемо кожен з методів фільтрації детальніше.

1.3.1. Фільтрація на основі вмісту

З назви підходу, можна зрозуміти, що фільтрація на основі вмісту перш за все базується на даних про елементи всередині системи, а не інформації про користувачів. Тобто, рекомендації формуються завдяки знаходженню схожих елементів, до тих, які користувач вже оцінював в минулому. Для цього створюється профіль елемента (реklamного оголошення) та профіль користувача. Після цього завдяки даним про елементи системи, можна побудувати відповідність конкретного елемента системи конкретному користувачу. Для того, щоб описати елементи системи та створити їх профілі, рекомендаційна система ставить у відповідність кожному елементу конкретний набір ключових слів, наприклад заголовок рекламного оголошення, категорії до яких належить рекламне оголошення. Роботу рекомендаційної системи на основі фільтрації вмісту зображено на рис. 2.

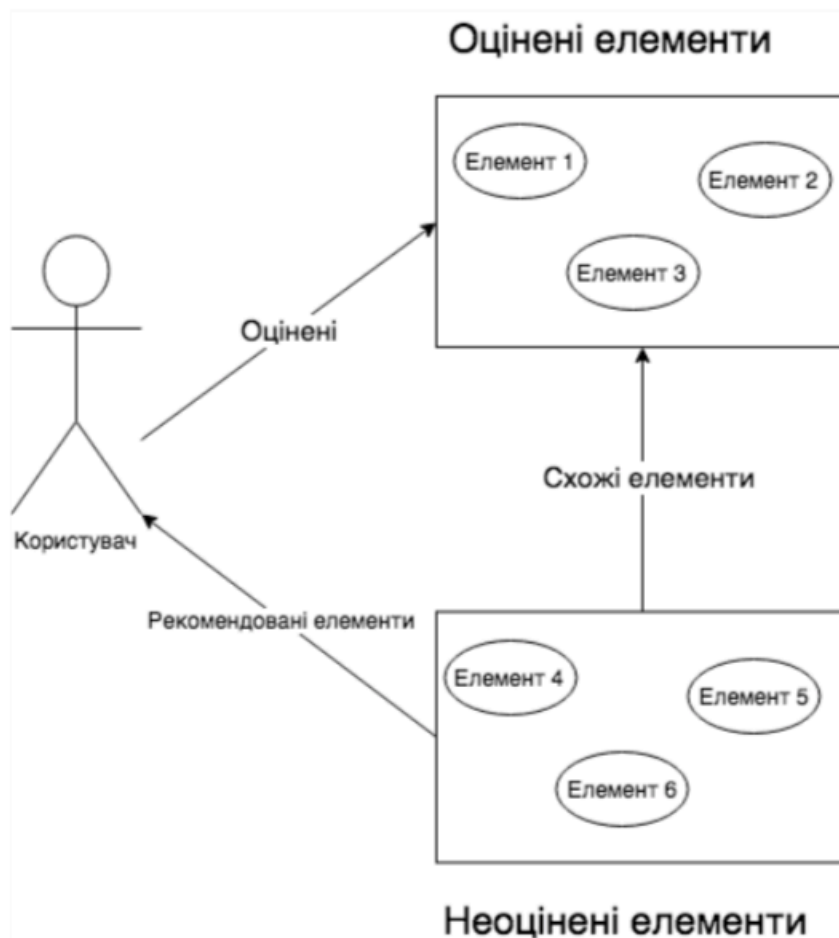


Рис. 2. Принцип за яким працює рекомендаційна система фільтрації на основі вмісту

Рекомендаційні системи з використанням фільтрації на основі вмісту використовують функцію зіставлення деякого користувача u з деяким елементом системи $s - h(u, s)$. Ця функція використовує дані про інші елементи системи $s' \in \mathcal{S}$, яким користувач вже давав оцінку в минулому, а також схожість цих елементів з конкретним елементом s .

Тобто, для того щоб система змогла рекомендувати користувачу нові елементи, вона повинна спершу провести аналіз елементів, які вже були оцінені користувачем у минулому. Після цього, потрібно знайти ознаки, які поєднують ці елементи, та, базуючись на знайдених ознаках, знайти такі елементи, які будуть відповідати цим ознакам, а також не будуть мати

оцінок від користувача (це означає, що вони будуть для нього новими елементами).

Якщо сказати більш простою мовою, якщо певний елемент А був позитивно оцінений користувачем u , тобто сподобався йому, а елемент В за певними ознаками схожий на елемент А, то можна зробити припущення, що з великою вірогідністю елемент В, також сподобається користувачу u .

Отже, профіль користувача формується у вигляді параметрів, що належать кожному елементу $s' \in S'$, де S' – це множина елементів, що були оцінені користувачем. Зазвичай, цими параметрами виступають ключові слова, а також вагові коефіцієнти ключових слів для кожного елемента системи. Найчастіше, для знаходження цих вагових коефіцієнтів, використовується спосіб, який називається TF-IDF міра [19].

TF-IDF міра обчислюється за такою формулою:

$$TF_{ij} = \frac{f_{ij}}{\max_z f_{zj}}, \quad (2)$$

де f_{ij} – це кількість входжень деякого ключового слова k_i в елемент d_j , при цьому ключове слово k_j зустрічається в n_i елементах системи.

Але при використанні формули 2 враховується тільки частота входження ключових слів. Це може призвести до випадку, коли максимальну вагу будуть мати найбільш розповсюджені ключові слова, що в подальшому призведе до неправильного прогнозування вподобань певного користувача. Для того, щоб уникнути такої ситуації, використовується величина IDF_i , яка є оберненою до частоти входження ключового слова в елемент (рекламне оголошення):

$$IDF_i = \log \frac{N}{n_i}, \quad (3)$$

де N – це кількість елементів системи, які потенційно можуть бути рекомендовані певному користувачу.

Виходячи з формули 2 та 3, маємо можливість обчислити ваговий коефіцієнт w_{ij} ключового слова k_i в елементі d_j за наступною формулою:

$$w_{ij} = TF_{ij} * IDF_i, \quad (4)$$

Профіль елемента d_j можна задати таким чином:

$$Content(d_j) = (w_{1j}, \dots, w_{kj}), \quad (5)$$

Рекомендаційні системи, які основані на фільтрації за вмістом, при формуванні рекомендацій оперують даними про елементи системи, які вже були оцінені користувачами в минулому. З множини усіх елементів наявних в системі – обираються тільки ті, що є найбільш схожими до елементів, які вже були оцінені. Таким чином, будується гіпотеза, що нові елементи також будуть добре оцінені користувачем. Набір елементів системи, що оцінив користувач, формують профіль користувача, а також ваговий вектор ключових слів:

$$(w_{u1}, \dots, w_{uk}), \quad (6)$$

де кожний ваговий коефіцієнт w_{ui} визначає важливість ключового слова k_i для користувача u .

Тоді, за допомогою TF-IDF векторів \vec{w}_s та \vec{w}_u можна представити профіль елемента та профіль користувача відповідно, при цьому функція зіставлення $h(u, s)$ може бути представлена у вигляді косинусу кута між векторами \vec{w}_s та \vec{w}_u [20]:

$$h(u, s) = \cos(\vec{w}_u, \vec{w}_s) = \frac{\vec{w}_u \cdot \vec{w}_s}{\|\vec{w}_u\| \|\vec{w}_s\|} = \frac{\sum_{i=1}^K w_{ui} w_{is}}{\sqrt{\sum_{i=1}^K w_{ui}^2} \sqrt{\sum_{i=1}^K w_{is}^2}}, \quad (7)$$

де K – це загальна кількість ключових слів у системі.

1.3.2. Колаборативна фільтрація

Рекомендаційні системи, які використовують колаборативну фільтрацію, у порівнянні з фільтрацією за вмістом для своєї роботи проводять аналіз наявної інформації про користувачів, а не про елементи всередині системи. Кожному користувачу ставиться у відповідність група користувачів схожих з ним за вподобаннями. На основі цього формується гіпотеза, що група користувачів, яка однаково оцінила певні елементи в минулому, скоріше за все, також однаково оцінить інші елементи в майбутньому.

Роботу рекомендаційної системи на основі колаборативної фільтрації зображено на рис. 3.

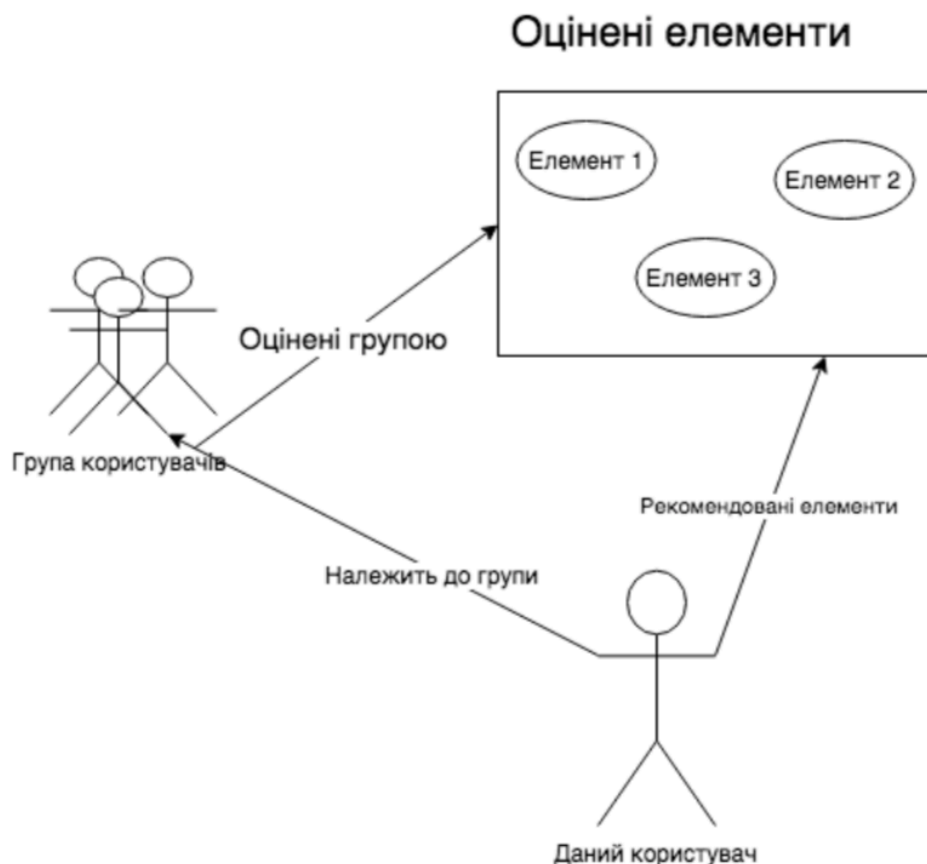


Рис. 3. Принцип за яким працює рекомендаційна система на основі колаборативної фільтрації

Тобто, рекомендаційні системи на основі методів колаборативної фільтрації, використовують дані про вже існуючі оцінки певних користувачів, які належать до однієї групи із даним користувачем за вподобаннями. В цьому випадку функція $h(u, s)$ задає ступінь зіставлення елемента s з користувачем u та обчислюється в залежності від зіставлення $h(u_j, s)$ для елемента s та для користувачів $u_j \in U$, де U – певна група користувачів [21].

Для того, щоб визначити міру схожості користувачів та сформувати групи користувачів, необхідно кожному користувачу створити профіль. Під час створення профіля можливе використання як даних про вподобання користувачів, так і загальної інформації такої як стать, вік, країна проживання, національність та інше. Такого роду інформація про користувача є дуже важливою на перших етапах роботи користувача з рекомендаційною системою для формування вподобань елементів його профіля, коли користувач ще не встиг оцінити велику кількість елементів системи. Коли рекомендаційна система має побудувати рекомендації новому користувачу про якого ще нічого не відомо, такого роду проблема називається “холодний старт”. Саме для того, щоб мінімізувати вплив проблеми холодного старту, необхідно використовувати загальні наявні дані про користувача, оскільки на основі цих даних користувача можна віднести до певної групи користувачів.

На сьогоднішній день існує дуже багато рекомендаційних систем, які працюють на основі принципу колаборативної фільтрації. Однією з перших таких систем, була “Grundy system” [22]. Ця рекомендаційна система являла собою програму, яка в процесі створення рекомендацій пропонувала користувачам відповісти на список певних запитань для того, щоб сформувати профіль користувача. Також, коли користувач надавав оцінку певному елементу системи, він повинен був дати відповідь на додаткові запитання для більшої обґрунтованості вибору. Наприклад, якщо користувач ставив погану оцінку певному елементу, система задавала

додаткові питання для того, щоб дізнатися причину такої оцінки, щоб в майбутньому уникнути рекомендації схожих елементів. В той самий час, інша система під назвою “Tapestry” [23] давала користувачам можливість самостійно знаходити інших користувачів системи, вподобання яких збігалися з їх вподобаннями.

Методи колаборативної фільтрації можна класифікувати на два типи:

- методи, які базуються на пам’яті;
- методи, які базуються на моделях.

Будемо розглядати та використовувати методи, які базуються на пам’яті.

Методи, які базуються на пам’яті, оперують даними про усі елементи системи, які були оцінені користувачем в минулому, для того, щоб передбачити рейтинг нового елементу:

$$r_{us} = \text{aggr} r_{u \in u', r_{u's}}, \quad (8)$$

де r_{us} – прогнозований рейтинг елемента s для користувача u , а u' – це множина користувачів, які в минулому оцінили елемент s , а також входять в одну групу з користувачем u .

В якості функції агрегування можна використовувати різні підходи. Найпростішим із існуючих підходів вважається загальне середнє значення:

$$r_{us} = \frac{1}{N} \sum_{u \in u'} r_{u's}, \quad (9)$$

де N – потужність множини U' .

Але частіше за все в якості функції агрегування використовується такий вираз:

$$r_{us} = k \sum_{u \in u'} \text{sim}(u, u') * r_{u's}, \quad (10)$$

де k – це параметр нормалізації, який обчислюється за формулою:

$$k = \frac{1}{\sum_{u \in u'} |\text{sim}(u, u')|}, \quad (11)$$

В формулі 11, функція $sim(u, u')$ – міра схожості. Ця величина є оберненою до відстані і найчастіше використовується як ваговий коефіцієнт. Тобто, в залежності від міри схожості користувачів u та u' , вплив оцінок наданих користувачем u' на формування рекомендацій для користувача u буде різною. У випадку, коли використовуються зважені суми, формула 10, неможливо враховувати деякі деталі. Наприклад, не усі користувачі системи оцінюють елементи всередині системи з однаково високим рівнем відповідальності. Для уникнення цього, використовується такий підхід [24]:

$$r_{us} = \vec{r}_u + k \sum_{u \in u'} sim(u, u') * (r_{u's} - \bar{r}_{u'}), \quad (12)$$

$$\bar{r}_u = \frac{1}{S_u} \sum_{s \in S_u} r_{us}. \quad (13)$$

Даний вираз включає в себе відмінності від загального рейтингу для певного користувача, замість того, щоб використовувати абсолютні оцінки.

Для того, щоб обчислити міру схожості $sim(u, u')$ користувачів, рекомендаційні системи використовують різноманітні підходи. Найчастіше схожість користувачів визначається саме за схожістю оцінок, які вони надають різним елементам системи. Для визначення схожості найчастіше використовуються такі 2 методи:

- знаходження кореляції;
- знаходження косинусних коефіцієнтів.

Нехай, $S_{uu'}$ – це множина всіх елементів, які були оцінені користувачами u та u' у минулому. Тоді, $S_{uu'}$ можемо задати таким чином:

$$S_{uu'} = \{s \in S \mid r_{u's} \neq 0, r_{us} \neq 0\}. \quad (14)$$

При колаборативній фільтрації $S_{uu'}$ використовується як проміжний параметр для того, щоб виявляти найбільш схожих між собою користувачів. Показник $S_{uu'}$ може обчислюватися багатьма різними

методами, найпростіший з них – це перетин множини S_u та $S_{u'}$. Деякі методи [25] дають можливість знайти найближчих сусідів для користувача u без кроку обчислення проміжного параметру $S_{uu'}$. Для цього використовують коефіцієнт Пірсона [26, 27]:

$$sim(u, u') = \frac{\sum_{s \in S_{uu'}} (r_{us} - \bar{r}_u)(r_{u's} - \bar{r}_{u'})}{\sqrt{\sum_{s \in S_{uu'}} (r_{us} - \bar{r}_u)^2 \sum_{s \in S_{uu'}} (r_{u's} - \bar{r}_{u'})^2}}, \quad (15)$$

В методах, що використовують косинусний коефіцієнт [28, 29], користувачі u та u' представляються як вектор в m -мірному просторі, де $m = S_{uu'}$. Отже, міра схожості між двома користувачами u та u' може бути обчислена за формулою:

$$sim(u, u') = \cos(\vec{u}, \vec{u'}) = \frac{\vec{u} \cdot \vec{u'}}{\|\vec{u}\| \|\vec{u'}\|} = \frac{\sum_{s \in S_{uu'}} r_{us} r_{u's}}{\sqrt{\sum_{s \in S_{uu'}} r_{us}^2 \sum_{s \in S_{uu'}} r_{u's}^2}}, \quad (16)$$

Варто зауважити, що косинусний коефіцієнт використовується в рекомендаційних системах на основі колаборативної фільтрації, так і в рекомендаційних системах на основі фільтрації за вмістом.

Існує велика кількість модифікацій та різноманітних вдосконалень, які збільшують ефективність роботи обговорених методів [30, 31]. Зумовлений рейтинг [32] є однією з модифікацій методу колаборативної фільтрації, який базується на пам'яті.

1.3.3. Гібридний метод

Деякі з рекомендаційних систем використовують суміш методів, створюючи гібридний підхід для формування рекомендацій. Головна ідея гібридного методу в рекомендаційних системах полягає у поєднанні результатів двох інших різних методів, наприклад фільтрації на основі вмісту та колаборативної фільтрації. Завдяки поєднанню двох методів, можливо уникнути недоліків поєднаних методів. Існують декілька основних варіантів комбінування декількох методів [18]:

- включення в колаборативну фільтрацію ознак елементів;
- включення в фільтрацію за вмістом ознак користувачів;
- побудова однієї моделі на основі фільтрації за вмістом та колаборативної фільтрації.

Розглянемо гібридний метод зі включенням в колаборативну фільтрацію ознак елементів. В першу чергу, такі системи працюють на основі використання методів колаборативної фільтрації, але вони включають в аналіз даних інформацію про елементи системи. Інформація про елементи системи, використовується для того, щоб визначити міру схожості між користувачами. Основною перевагою методу формування рекомендацій, полягає в тому, що підвищується релевантність сформованих рекомендацій в ситуації, коли у системі замала кількість пар користувачів з достатньою кількістю спільно оцінених елементів системи. Також, такий підхід дозволяє надавати користувачам рекомендації елементів, які схожі на інші елементи системи, а не тільки на основі вподобаннях інших схожих користувачів.

Розглянемо гібридний метод зі включенням в фільтрацію за вмістом ознак користувачів. Один з відомих методів реалізації гібридного методу є зниження розмірності профілів, які використовуються у фільтрації за вмістом. Така модифікація буде формувати рекомендації не тільки на основі фільтрації за вмістом, а також буде використовувати вподобання користувачів.

Розглянемо варіант побудови рекомендаційної системи на основі фільтрації за вмістом та колаборативної фільтрації. Цей гібридний метод формування рекомендацій на даний момент є найбільш популярним. Основна суть методу полягає в тому, що в одній системі застосовують характеристики фільтрації за вмістом та колаборативної фільтрації. Одним із варіантів реалізації такого методу є метод уніфікованих ймовірностей [34], який базується на роботі латентно-семантичного аналізу [35].

1.4. Висновки до розділу

В розділі було проведено огляд та аналіз існуючих моделей та методів, які використовуються в рекомендаційних системах для формування рекомендацій, а саме було розглянуто такі фільтрації:

- фільтрацію за вмістом;
- колаборативну фільтрація;
- гібридний метод фільтрації.

Після проведення огляду існуючих методів є актуальним розробка модифікованого методу формування рекомендацій.

2. МОДИФІКОВАНИЙ МЕТОД ФОРМУВАННЯ РЕКОМЕНДАЦІЙ

2.1. Модифікований метод формування рекомендацій

Запропонований у роботі модифікований метод формування рекомендацій рекламних оголошень буде полягати у застосуванні трьох різних методів для формування рейтингів елементів системи.

Спочатку необхідно розділити колаборативний метод формування рекомендацій на 2 окремих:

- фільтрація на основі сусідства;
- демографічна фільтрація.

В методі фільтрації на основі сусідства рейтинг певного елементу s для користувача u формується за рахунок рейтингів користувачів, що вже мають вподобання схожі на думку системи із вподобаннями даного користувача. Ефективність роботи цього методу залежить в основному від обраної кількості сусідів для користувача u , оцінки яких і будуть враховуватися при формуванні рейтингу для користувача. Ступінь сусідства користувачів обчислюється за допомогою міри схожості, а саме кореляції або косинусної міри.

Для надання рекомендацій при створенні груп користувачів важливу роль відіграють їх демографічні характеристики. Більшість рекомендаційних систем дуже часто натикаються на проблему холодного старту через малу кількість даних про наявні оцінки користувачів [36], про яку йшлося вище. В цих випадках є необхідним використання даних, які доступні з самого початку роботи користувача з системою. Демографічні дані користувача відносяться саме до такого типу даних, адже у випадку рекламних оголошень їх завжди можна буде отримати з інформації браузера користувача. Це наповнить профіль користувача мінімальними доступними даними на початку роботи з системою. Завдяки цим даним, система зможе сформувати початкові рекомендації з більшою точністю.

Маючи інформацію про країну, національність, можливо вік та стать користувача, набагато легше віднести його до певної групи користувачів.

Основною задачею рекомендаційних систем є передбачення рейтингу елементів системи (реklamних оголошень). В створеному модифікованому методі прогноз рейтингу елементів буде обчислюватися за рахунок поєднання трьох методів, описаних вище: фільтрації за вмістом, фільтрації на основі сусідства та демографічної фільтрації.

Кожен метод буде незалежно від інших методів обчислювати прогнозований рейтинг для елементів системи. Після отримання результатів усіх методів, потрібно лінійно об'єднати результати, окремо врахувавши внесок кожного методу в отриманні фінального рейтингу.

Розглянемо приклад: якщо користувач u вже надавав оцінки певному набору елементів, схожих на елемент s , тоді буде доцільним підвищити вагу результату, який утриманий за допомогою методу фільтрації за вмістом. І навпаки, якщо користувач u має дуже велику кількість сусідів, які вже оцінили елемент s у минулому, то більшу вагу повинен буде мати метод фільтрації на основі сусідства.

Варто зазначити, що у випадку холодного старту, найбільшу вагу результату буде мати демографічна фільтрація, так як в системі будуть майже відсутні дані про вподобання користувача.

Для обчислення міри впливу отриманого результату кожного із методів, потрібно ввести коефіцієнт, який буде задавати вагу цього результату.

Тоді фінальний прогнозований рейтинг буде обчислюватися за такою формулою:

$$r_{us} = \frac{\alpha r_{us}^{DF} + \beta r_{us}^{CD} + \gamma r_{us}^{CF}}{\alpha + \beta + \gamma}, \quad (17)$$

де r_{us}^{DF} – це прогнозований рейтинг для елемента s для користувача u , який був отриманий за допомогою метода демографічної фільтрації, r_{us}^{CD} – це прогнозований рейтинг для елемента s для користувача u , який був

отриманий за допомогою метода колаборативної фільтрації, r_{us}^{CF} – це прогнозований рейтинг для елемента s для користувача u , який був отриманий за допомогою метода фільтрації на основі вмісту, α , β , γ – це вагові коефіцієнти для кожного з методів відповідно.

Внесок результатів оцінок кожного з алгоритмів у фінальний результат буде залежати від значення вагового коефіцієнта для цього метода. Зі збільшенням кількості наявних оцінок від користувачів, потрібно збільшувати значення вагового коефіцієнта для метода колаборативної фільтрації. І навпаки, якщо в системі мала кількість оцінок від користувачів, або у випадку проблеми холодного старту, варто збільшувати внесок результатів роботи методів фільтрації за вмістом та демографічної фільтрації.

Для того, щоб обчислювати значення цих вагових коефіцієнтів, потрібно використати деяку функцію $S(x)$, де x – це кількість оцінок в середині системи від користувачів, яка буде повертати максимальне значення 1 для великої кількості оцінок, та 0 для малої кількості оцінок відповідно. Таким умовам підходить функція сигмоїда, яка задається за допомогою логістичної рівності:

$$S(x) = \frac{1}{1 + e^{-x}} = \frac{e^x}{e^x + 1}. \quad (18)$$

Графік цієї функції буде виглядати наступним чином (рис. 4).

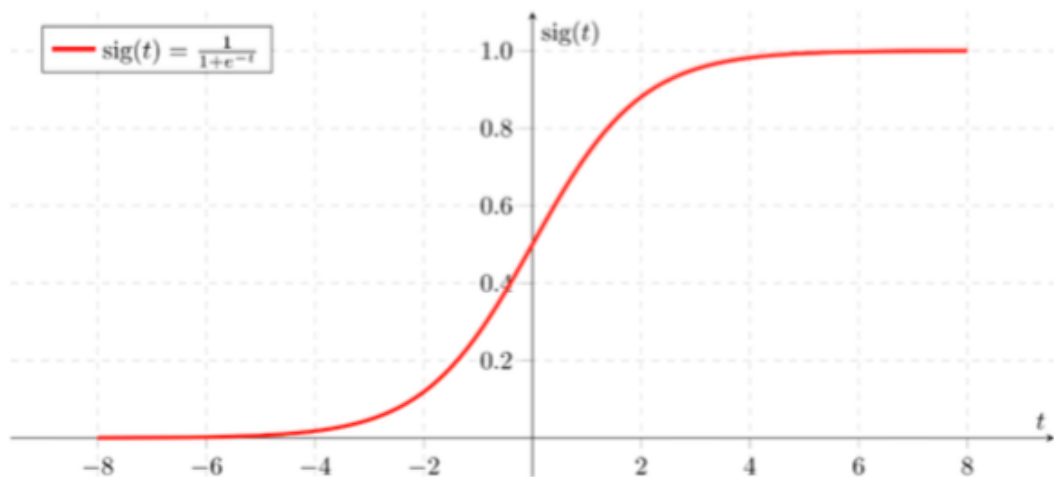


Рис. 4. Графік функції $S(x)$

За допомогою цієї функції будемо задавати вагові коефіцієнти використовуючи систему рівнянь:

$$\begin{aligned} \gamma &= \frac{1}{1 + e^{\frac{-x}{2}}}, \\ \alpha = \beta &= 1 - \frac{1}{1 + e^{\frac{-x}{2}}}, \end{aligned} \quad (20)$$

де x – це кількість наданих оцінок в системі від користувачів, значення параметрів α , β , γ будуть змінюватися динамічним шляхом, в залежності від наявної інформації про кількість оцінок від користувачів в середині системи.

2.2. Опис модифікованого метода

Представимо запропонований модифікований метод у вигляді послідовних кроків роботи алгоритму:

1. Проведення демографічної фільтрації користувачів системи:

- 1.1. Обчислення вагових коефіцієнтів для кожного з користувачів на основі подібності демографічних даних з активним користувачем за допомогою міри схожості.

- 1.2. Обрання з усіх користувачів системи групи певного розміру, які найбільш схожі на активного користувача.
 - 1.3. Обчислення прогнозованого рейтингу елементів для активного користувача на основі вже існуючих оцінок користувачів із обраної групи (формула 10).
2. Проведення фільтрації за вмістом для елементів системи (оголошень):
 - 2.1. Обчислення вагових коефіцієнтів для кожного з елементів системи на основі подібності ознак елементів системи (заголовки, ключові слова, категорія та інші) з заданим елементом за допомогою використання міри схожості.
 - 2.2. Обрання з усіх елементів групи елементів найбільш подібних до даного елементу.
 - 2.3. Обчислення прогнозованого рейтингу кожного елемента системи на основі наявних оцінок інших елементів системи (формула 7).
3. Проведення колаборативної фільтрації:
 - 3.1. Обчислення вагових коефіцієнтів для кожного з користувачів на основі подібності їх вподобань із вподобаннями активного користувача за допомогою використання міри схожості.
 - 3.2. Обрання з усіх користувачів системи групи певного розміру, яка містить схожих користувачів на активного користувача.
 - 3.3. Обчислення прогнозованого рейтингу елементів на основі вже існуючих оцінок користувачів з обраної групи (формула 10).
4. Обчислення вагових коефіцієнтів для демографічного, колаборативного методів фільтрації та фільтрації на основі вмісту.

5. Обчислення фінального прогнозованого рейтингу елементів системи (формула 17).

Отримати результати роботи трьох окремих методів можна трьома окремими потоками, схема алгоритму також відображена на рис. 5.

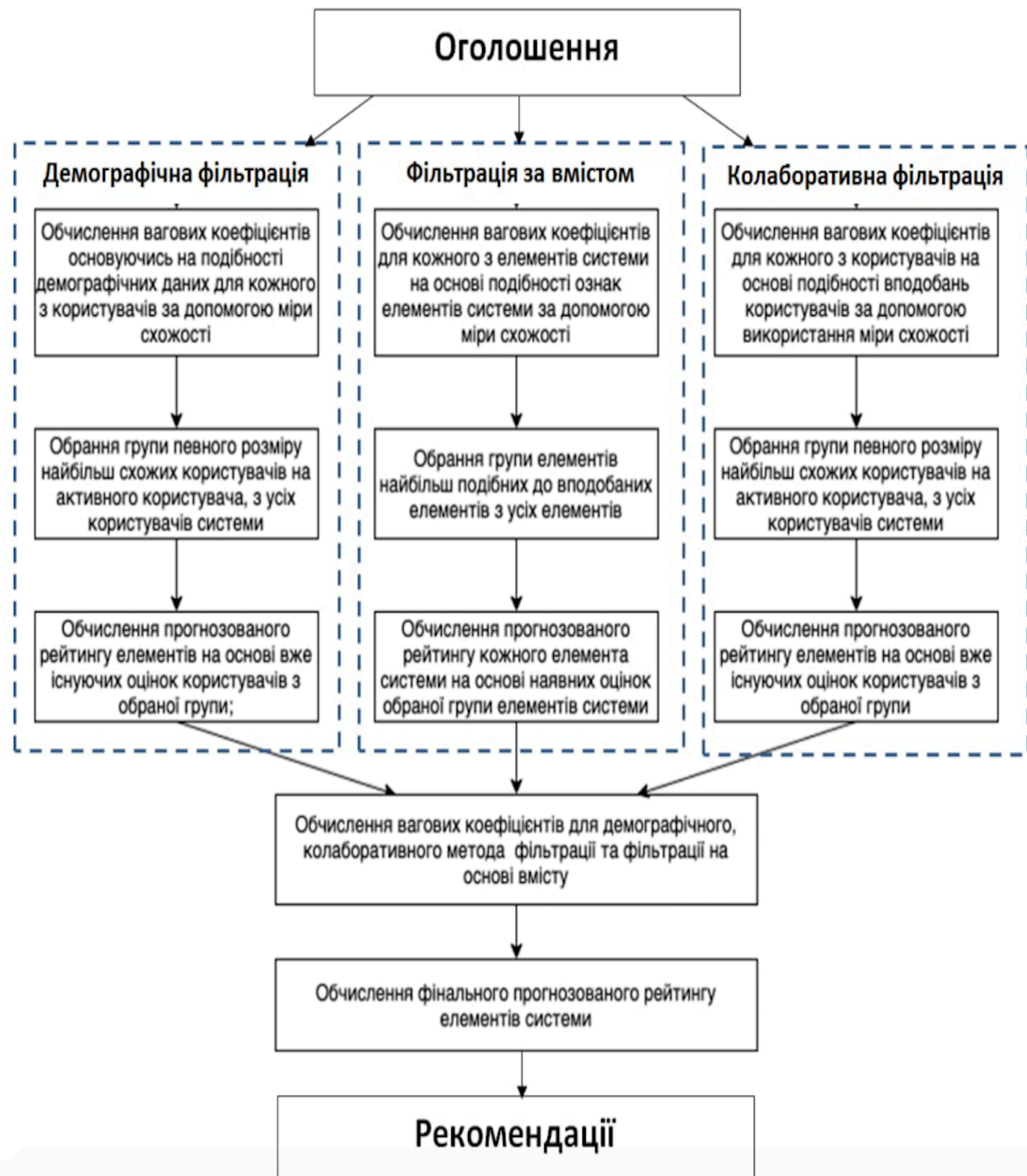


Рис. 5. Схема алгоритму роботи модифікованого методу

2.3. Висновки до розділу

В даному розділі описано розроблений модифікований гібридний метод формування рекомендацій рекламних оголошень з використання колаборативної фільтрації, демографічної фільтрації та фільтрації на основі змісту. Також було наведено алгоритм роботи розробленого модифікованого гібридного метода формування рекомендацій.

3. АНАЛІЗ ПРОГРАМНОЇ РЕАЛІЗАЦІЇ

3.1. Засоби розробки програмного забезпечення

Розроблений програмний метод формування рекомендацій на основі аналізу вподобань користувачів є однією з частин рекламної платформи.

При розробці даного методу формування рекомендацій були використані наступні засоби:

JavaScript [15] – це об'єктно-орієнтована, прототипна та динамічна мова програмування, реалізація стандарту ECMAScript. Цю мову класифікують як прототипну (підмножина об'єктно-орієнтованої), скриптову з динамічною типізацією. Також частково підтримує інші парадигми програмування і деякі відповідні архітектурні властивості, такі як: динамічна та слабка типізація, автоматичне керування пам'яттю, функції як об'єкти першого класу, прототипне наслідування. JavaScript має багато властивостей притаманних об'єктно-орієнтованим мовам, але завдяки концепції прототипів підтримка об'єктів в ній відрізняється від традиційних мов ООП. Крім того, JavaScript має ряд властивостей, притаманних функціональним мовам, – функції як об'єкти першого класу, об'єкти як списки, анонімні функції, замикання – що додає мові додаткову гнучкість. JavaScript має C-подібний синтаксис. Усі браузері інтерпретують код JavaScript на веб-сторінках.

Node.js [11] – це асинхронне, подієве оточення для мови JavaScript. Node.js було створено для побудови масштабних мережесхем додатків. Він може одночасно обробляти велику кількість з'єднань, для кожного з'єднання викликається функція зворотнього виклику, але якщо немає жодного з'єднання, то Node.js засинає. Це відрізняється від загальної моделі обробки з'єднань, в якій використовують паралельні потоки операційної системи. Також можна не турбуватися про блокування процесів, так як їх немає. Оскільки нічого не блокується, то за допомогою node.js дуже легко створювати масштабовані системи. Node.js не має

багатопоточності, але замість цього можна використовувати можливості декількох ядер.

HTML [9] – мова розмітки, яка створена для розробки веб-сторінок. Це стандарт, який використовується для створення вмісту веб-сторінок, визначає структуру та код для визначення вмісту веб-сторінки, наприклад, тексту, зображення, відео, ігри, серед інших. Це стандарт, що застосований для відображення веб-сторінок і прийнятий всіма поточними браузерами.

CSS [7] — це мова стилів, яка використовується для візуалізації веб-документів та інтерфейсів, які написані на мові HTML. Також мова CSS може бути застосована до будь-якого документа типу XML.

MongoDB [10] – це документо-орієнтована система керування базами даних з відкритим вихідним кодом. Вона не потребує опису схеми таблиць. MongoDB займає нішу між швидкими і масштабованими системами, що оперують даними у форматі ключ/значення, і реляційними СКБД. MongoDB підтримує зберігання документів в JSON-подібному форматі, має гнучку мову для формування запитів, повністю підтримує індекси, реплікацію. У MongoDB є вбудовані засоби із забезпечення шардінгу. MongoDB підтримується багатьма мовами програмування, легка у використанні та налаштуванні. Основні переваги MongoDB: відсутність схеми таблиць, легкість, масштабованість, швидкість, збереження даних у вигляді JSON-об'єктів, безпечність, відкритий код, вичерпна документація

Для методу формування рекомендацій рекламних оголошень може підійти як реляційна база так і документо-орієнтована MongoDB. Також, за допомогою Node.js модуля mongoose, MongoDB можна використовувати, як реляційну базу, так як цей модуль потребує опису усіх полів сутностей. Було зроблено вибір, використовувати MongoDB, тому що однією з основних переваг є JSON формат документів, це зекономить час під час роботи з базою. Також MongoDB легко масштабується при виникненні потреби, та не програє у швидкодії іншим реляційним базам даних.

3.2. Структура бази даних

При розробці системи, будемо використовувати базу даних MongoDB, як реляційну базу даних. Спроекуємо базу даних у концептуальному вигляді за допомогою ER-діаграми (рис. 6).

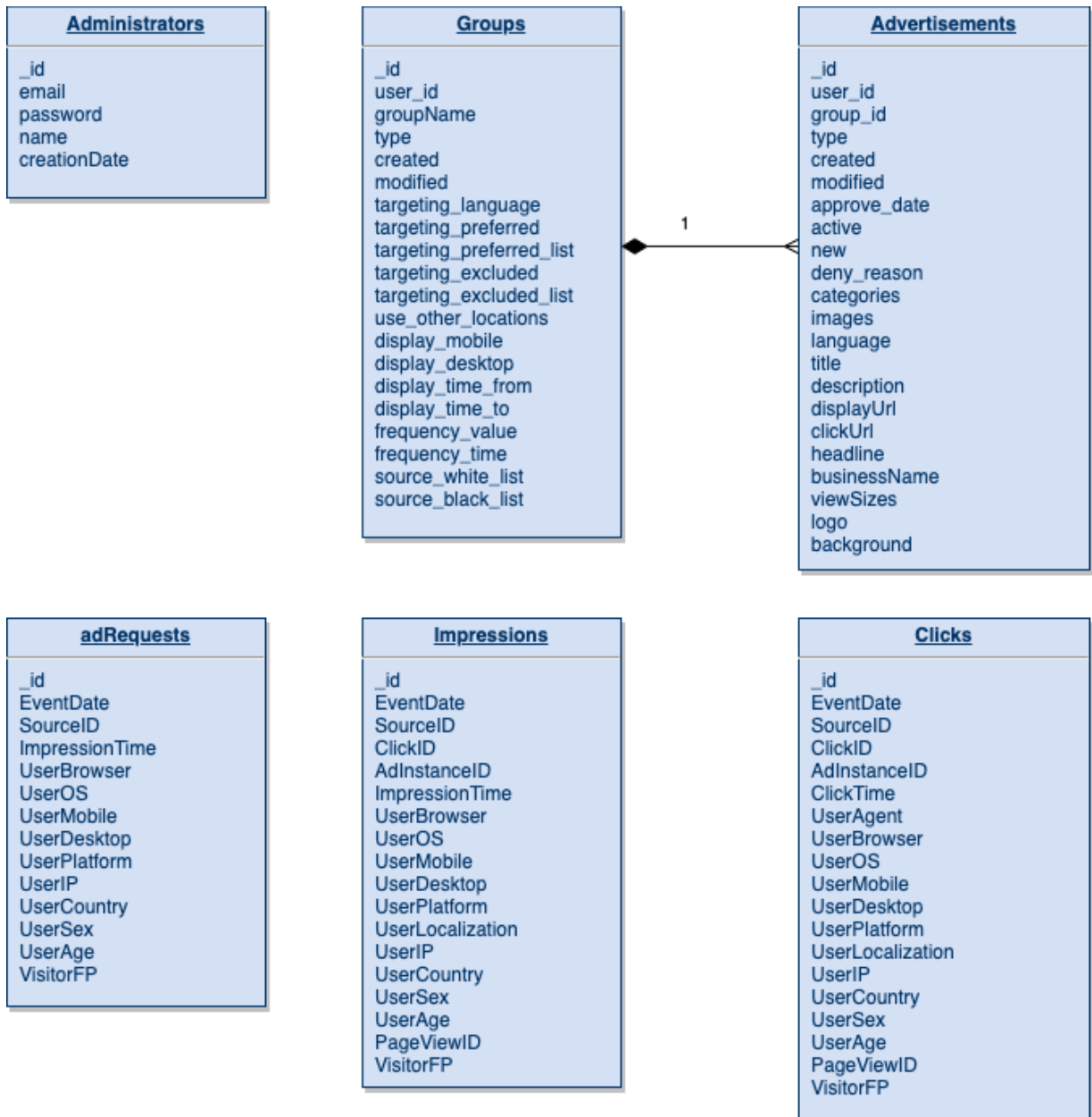


Рис. 6. ER-діаграма бази даних

Опишемо таблиці, які будуть використовуватися для зберігання та обробки даних в середині системи, та які відображені на ER-діаграмі за допомогою таблиці 1.

Таблиці бази даних

Назва таблиці	Опис
Administrators	Таблиця, що містить інформацію про адміністраторів системи
Groups	Таблиця, що містить інформацію про групи рекламних оголошень
Advertisements	Таблиця, що містить інформацію про рекламні оголошення
AdRequests	Таблиця, що містить інформацію про запити на отримання рекламного оголошення
Impressions	Таблиця, що містить інформацію про перегляди рекламних оголошень
Clicks	Таблиця, що містить інформацію про переходи за посиланням рекламних оголошень

Наведемо детальний опис кожної з таблиць бази даних, що використовуються.

Опис полів таблиці Administrators наведено у таблиці 2.

Таблиця 2

Таблиця бази даних Administrators

Назва поля	Тип	Обмеження	Опис
_id	ObjectId	Не може бути порожнім	Унікальний ідентифікатор
email	String	Не може бути порожнім	Email адміністратора
password	String	Не може бути порожнім	Пароль
name	String		Ім'я

creationDate	Date	Не може бути порожнім	Дата створення запису адміністратора
--------------	------	-----------------------	--------------------------------------

Опис полів таблиці Groups (групи рекламних оголошень) наведено у таблиці 3.

Таблиця 3

Таблиця бази даних Groups

Назва поля	Тип	Обмеження	Опис
_id	ObjectId	Не може бути порожнім	Унікальний ідентифікатор групи
user_id	ObjectId	Не може бути порожнім	Унікальний ідентифікатор юзера, якому належить реклама
groupName	String	Не може бути порожнім	Назва групи рекламних оголошень
type	String	Не може бути порожнім	Тип групи
created	Date	Не може бути порожнім	Дата створення
modified	Date		Дата оновлення
targeting_language	String	Не може бути порожнім	Мова таргетингу
locations_preferred	Boolean		Чи показувати рекламні оголошення в обраних країнах

locations_preferred_list	Array[String]		Список обраних країн в яких показувати рекламне оголошення
locations_excluded	Boolean		Чи не показувати рекламні оголошення в обраних країнах
locations_excluded_list	Array[String]		Список обраних країн в яких не показувати рекламні оголошення
use_other_locations	Boolean		Чи показувати в інших країнах
display_mobile	Boolean	Не може бути порожнім	Чи показувати на телефонах
display_desktop	Boolean	Не може бути порожнім	Чи показувати на комп'ютерах
display_time_from	Number	Не може бути порожнім	З якого часу дня показувати рекламу
display_time_to	Number	Не може бути порожнім	До якого часу дня показувати рекламу
frequency_value	Number		Скільки разів можна показати одне рекламне оголошення одному користувачу

frequency_time	Number		Проміжок часу за який можна показати рекламу декілька разів
source_white_list	Array[String]		Список сайтів на яких показувати рекламні оголошення
source_black_list	Array[String]		Список сайтів на яких не показувати рекламні оголошення

Опис полів таблиці Advertisements (реklamні оголошення) наведено у таблиці 4.

Таблиця 4

Таблиця бази даних Advertisements

Назва поля	Тип	Обмеження	Опис
_id	ObjectId	Не може бути порожнім	Унікальний ідентифікатор групи
user_id	ObjectId	Не може бути порожнім	Унікальний ідентифікатор юзера, якому належить реклама
group_id	ObjectId	Не може бути порожнім	Унікальний ідентифікатор групи, до якої належить рекламне оголошення

Продовження табл. 4

type	String	Не може бути порожнім	Тип рекламного оголошення
created	Date	Не може бути порожнім	Дата створення
modified	Date		Дата оновлення
approved	Boolean	Не може бути порожнім	Чи дозволене рекламне оголошення для показу
approve_date	Date		Дата проходження перевірки рекламним оголошенням
active	Boolean	Не може бути порожнім	Чи запущене рекламне оголошення
new	Boolean		Чи не показувати рекламні оголошення в обраних країнах
deny_reason	String		Причина у відмові показувати рекламне оголошення
categories	Array[String]	Не може бути порожнім	Список категорій до яких належить рекламне оголошення

Продовження табл. 4

images	Array[String]	Не може бути порожнім для картинного типу рекламного оголошення	Рекламні картинки
language	String	Не може бути порожнім	Мова реклами
title	String	Не може бути порожнім	Заголовок рекламного оголошення
description	String	Не може бути порожнім для текстового типу рекламного оголошення	Опис для рекламного оголошення
displayUrl	String	Не може бути порожнім для текстового типу рекламного оголошення	Url який відображати користувачу
clickUrl	String	Не може бути порожнім	Url за яким перейде користувач при натисканні на рекламне оголошення
headline	String	Не може бути порожнім для універсального типу рекламного оголошення	Заголовок рекламного оголошення

Продовження табл. 4

businessName	String	Не може бути порожнім для універсального типу рекламного оголошення	Назва фірми для рекламного оголошення
viewSizes	Array[String]		Розміри в яких показувати рекламне оголошення
logo	Mixed	Не може бути порожнім для універсального типу рекламного оголошення	Логотип для рекламного оголошення
background	Mixed	Не може бути порожнім для універсального типу рекламного оголошення	Фон для рекламного оголошення

Опис полів таблиці adRequests (запити на рекламні оголошення) наведено у таблиці 5.

Таблиця 5

Таблиця бази даних adRequests

Назва поля	Тип	Обмеження	Опис
_id	ObjectId	Не може бути порожнім	Унікальний ідентифікатор групи
EventDate	Date	Не може бути порожнім	Дата запиту

Продовження табл. 5

SourceID	String	Не може бути порожнім	Унікальний ідентифікатор сайту, з якого прийшов запит
ImpressionsTime	Date	Не може бути порожнім	Дата та час перегляду
UserBrowser	String		Браузер користувача
UserOs	String		Операційна система користувача
UserMobile	Number		Чи прийшов запит з телефону
UserDesktop	Number		Чи прийшов запит з комп'ютера
UserPlatform	String		Платформа браузера користувача
UserIP	String		IP адреса користувача
UserCountry	String		Країна запиту користувача
UserSex	String		Стать користувача
UserAge	Number		Вік користувача
VisitorFP	String		Унікальний цифровий відбиток користувача

Опис полів таблиці Impressions (перегляди рекламних оголошень) наведено у таблиці 6.

Таблиця бази даних Impressions

Назва поля	Тип	Обмеження	Опис
_id	ObjectId	Не може бути порожнім	Унікальний ідентифікатор групи
EventDate	Date	Не може бути порожнім	Дата запиту
ClickID	String		Унікальний ідентифікатор кліку користувача
SourceID	String	Не може бути порожнім	Унікальний ідентифікатор сайту, з якого прийшов запит
AdInstanceID	String	Не може бути порожнім	Унікальний ідентифікатор рекламного блока з якого був зроблений запит
ImpressionsTime	Date	Не може бути порожнім	Дата та час перегляду
UserAgent	String		Юзер агент користувача у браузері
UserBrowser	String		Браузер користувача
UserOs	String		Операційна система користувача
UserMobile	Number		Чи прийшов запит з телефону
UserDesktop	Number		Чи прийшов запит з комп'ютера

Продовження табл. 6

UserPlatform	String		Платформа браузера користувача
UserLocalization	String		Мова браузера користувача
UserIP	String		IP адреса користувача
UserCountry	String		Країна запиту користувача
UserSex	String		Стать користувача
UserAge	Number		Вік користувача
PageViewID	String		Унікальний ідентифікатор перегляду сторінки з рекламним оголошенням
VisitorFP	String		Унікальний цифровий відбиток користувача

Опис полів таблиці Clicks (переходи за посиланням рекламних оголошень) наведено у таблиці 7.

Таблиця 7

Таблиця бази даних Clicks.

Назва поля	Тип	Обмеження	Опис
_id	ObjectId	Не може бути порожнім	Унікальний ідентифікатор групи
EventDate	Date	Не може бути порожнім	Дата запиту

ClickID	String		Унікальний ідентифікатор кліку користувача
SourceID	String	Не може бути порожнім	Унікальний ідентифікатор сайту, з якого прийшов запит
AdInstanceID	String	Не може бути порожнім	Унікальний ідентифікатор рекламного блока з якого був зроблений запит
ClickTime	Date	Не може бути порожнім	Дата та час переходу
UserAgent	String		Юзер агент користувача у браузері
UserBrowser	String		Браузер користувача
UserOs	String		Операційна система користувача
UserMobile	Number		Чи прийшов запит з телефону
UserDesktop	Number		Чи прийшов запит з комп'ютера
UserPlatform	String		Платформа браузера користувача
UserLocalization	String		Мова браузера користувача
UserIP	String		IP адреса користувача

UserCountry	String		Країна запиту користувача
UserSex	String		Стать користувача
UserAge	Number		Вік користувача
PageViewID	String		Унікальний ідентифікатор перегляду сторінки з рекламним оголошенням
VisitorFP	String		Унікальний цифровий відбиток користувача

3.3. Архітектура розробленого програмного забезпечення

3.3.1. Діаграма класів

При розробці програмного забезпечення було використано ORM підхід роботи з базою даних за допомогою бібліотеки mongoose.

ORM – це технологія програмування, яка зв'язує бази даних з концепціями об'єктно-орієнтованих мов програмування, за допомогою створення «віртуальної об'єктної бази даних».

Mongoose дає можливість створювати моделі для документів бази даних та встановлювати взаємозв'язки між ними.

У нашому випадку таблицю в базі даних представлено певним класом, де один рядок таблиці – це екземпляр класу, а стовпчик – атрибут класу.

Це дуже зручний спосіб для взаємодії з базою даних, при цьому, вся бізнес-логіка описана у моделі.

Таким чином, діаграма класів системи, має майже такий самий вигляд, як ER-діаграма.

Класи в середині системи та зв'язки між ними зображені на рис. 7 та 8.

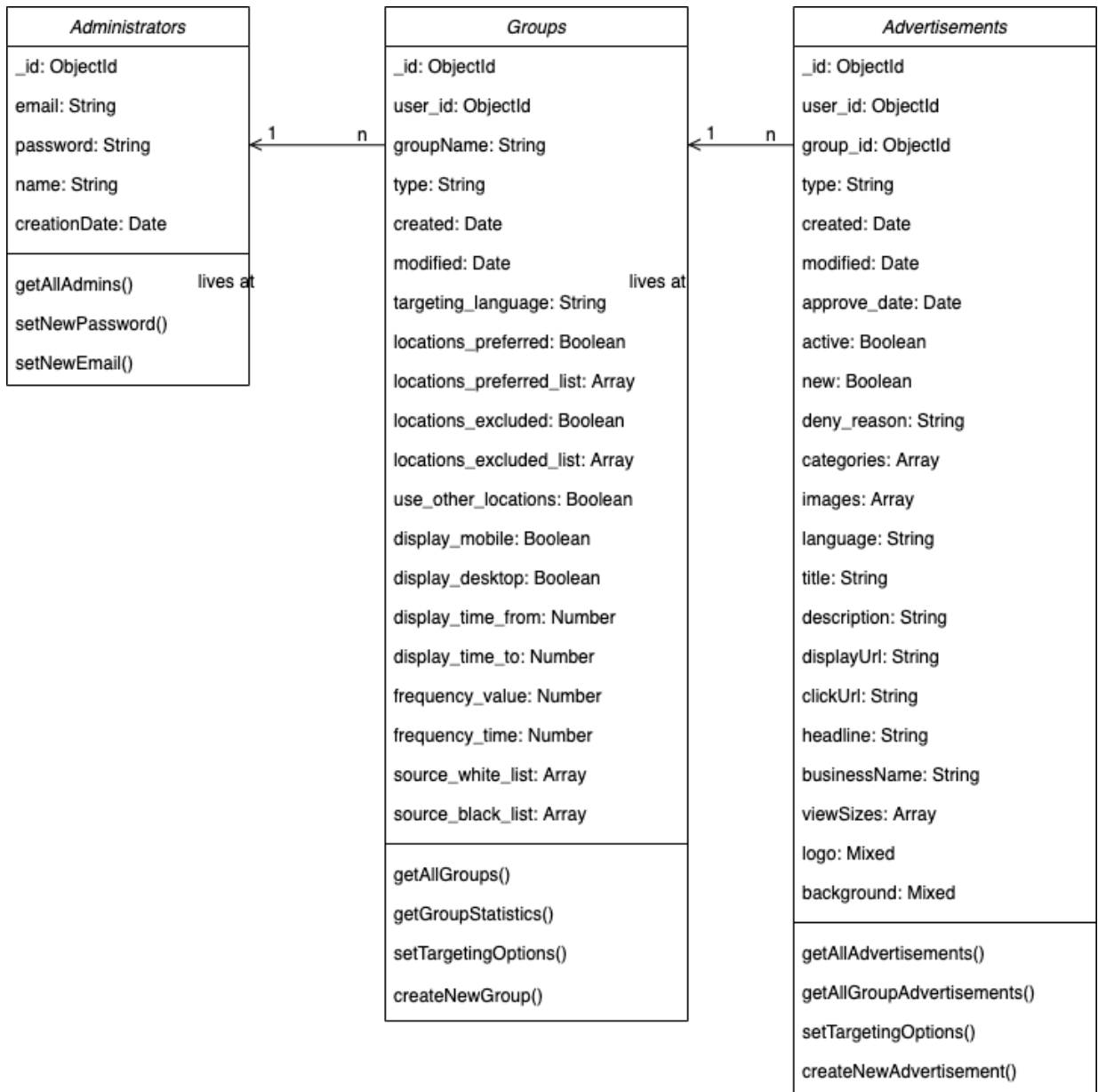


Рис. 7. Діаграма класів. Частина 1

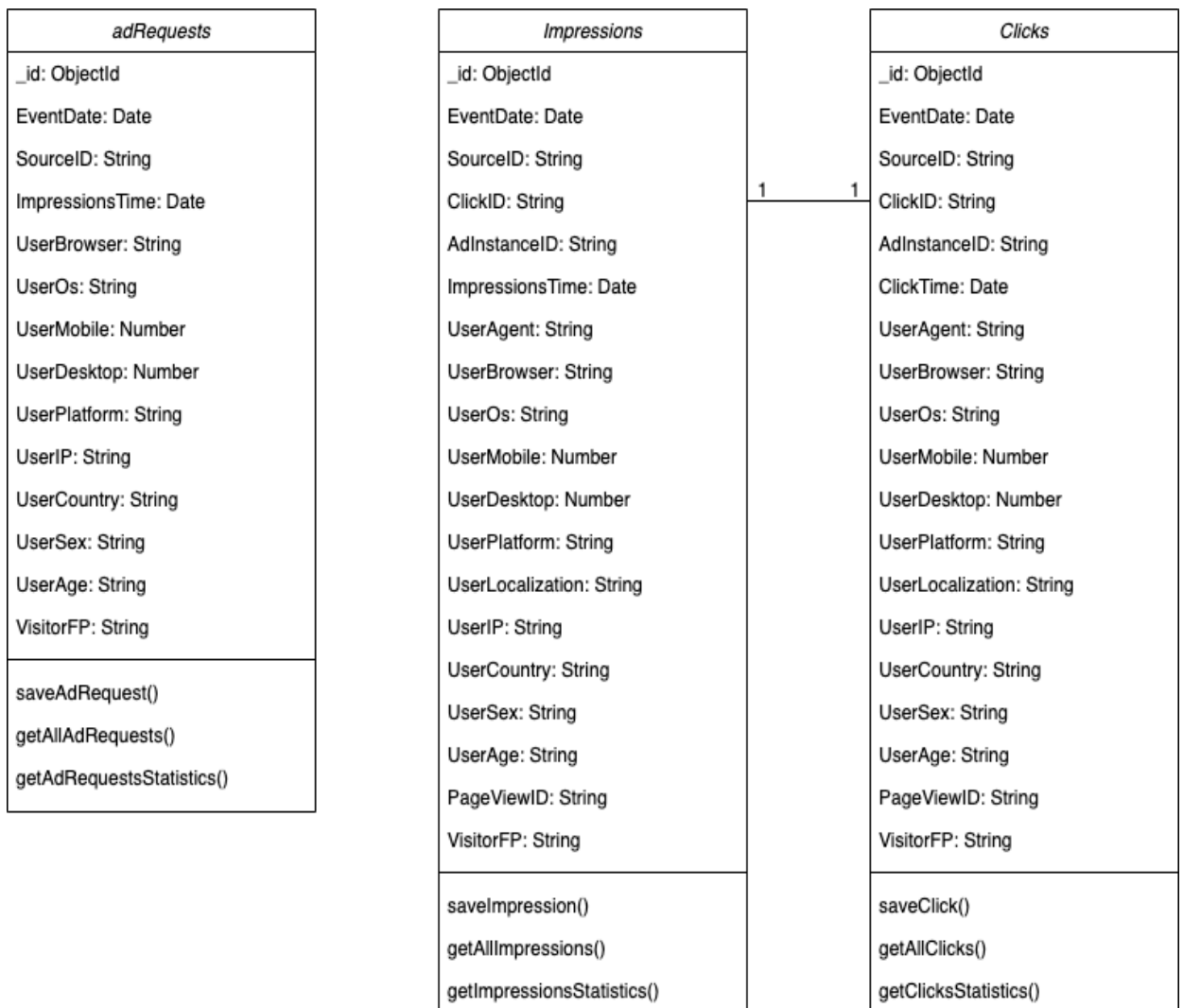


Рис. 8. Діаграма класів. Частина 2

3.3.2. Діаграма послідовності

Відповідальність кожного з компонентів системи описана у таблиці 8.

Таблиця 8

Відповідальність компонентів системи

Компонент	Відповідальність
Користувач	Перегляд рекламних оголошень на інтернет сайтах

Система	Виконує усю логіку роботи системи. Обробляє запити на отримання реклами, містить в собі роботу рекомендаційної систем і відповідає за взаємодію з базою даних.
Сервер бази даних	Сервер бази даних надає доступ до збереження та редагування даних. Обробляє запити, що надходять з серверу.

За допомогою діаграми послідовності зобразимо роботу системи на рис. 8 та у додатках.



Рис. 8. Діаграма послідовності

3.4. Опис технічного забезпечення

Компоненти системи взаємодіють одне з одним за допомогою https протоколу. Діаграма взаємодії компонентів системи зображена на рис. 9.

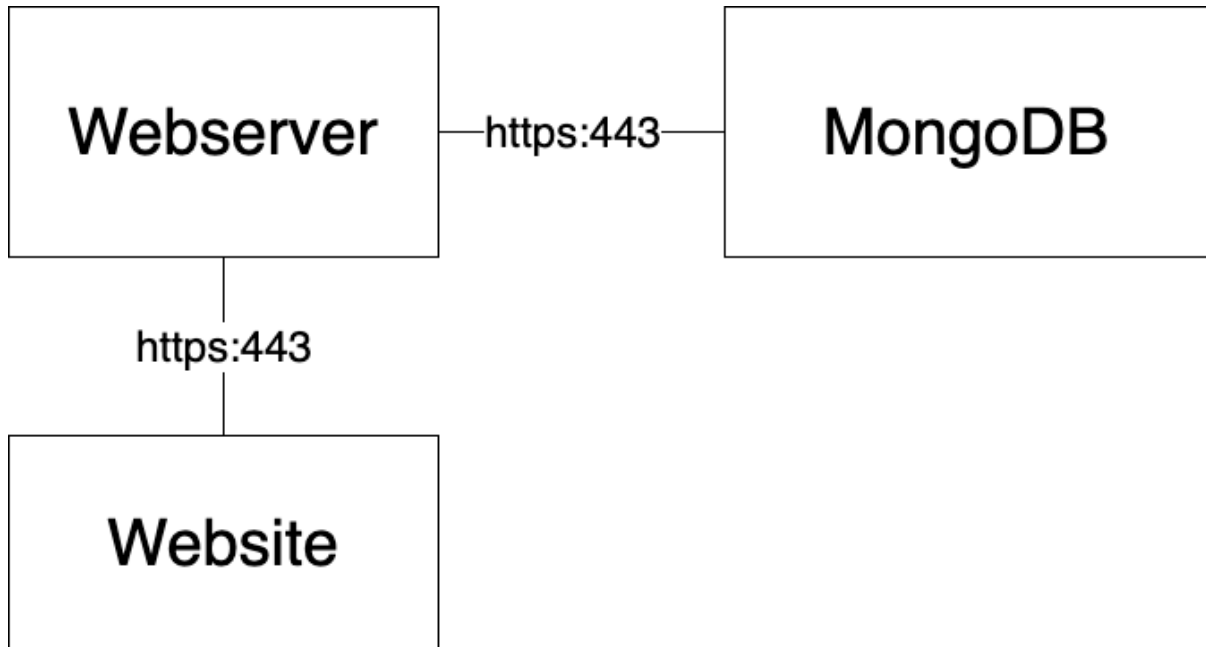


Рис. 9. Діаграма взаємодії компонентів

Для коректної роботи розроблюваної серверної частини системи, який містить метод формування рекомендацій, рекомендовано використовувати сервер, з наступними характеристиками:

- 4-ядерний процесор з частотою не менш ніж 2.0 ГГц;
- оперативна-пам'ять формату DDR5 та об'ємом не менш ніж 16 ГБ;
- твердотілий накопичувач (SSD) об'ємом не менш, ніж 128 Гб;
- безперервний доступ до мережі Інтернет.

3.5. Висновки до розділу

У даному розділі було детально розглянуто архітектуру розробленого програмного забезпечення, а саме:

- описано загальну архітектуру ПЗ;
- описано структуру бази даних за допомогою ER-діаграми;

- описано таблиці бази даних, а також усі поля таблиць;
- описано діаграму класів системи;
- описано діаграму послідовності роботи системи;
- описано діаграму взаємодії компонентів системи.

4. АНАЛІЗ ОТРИМАНИХ РЕЗУЛЬТАТІВ

4.1. Тестування програмного продукту

Тестування програмного забезпечення є одним із важливих етапів створення програмного забезпечення.

Тестування програмного забезпечення – це процес, під час якого перевіряється відповідність заявлених на початку вимог до продукту або до результатів роботи продукту і реально реалізованих функціональних можливостей продукту або отриманих результатів роботи відповідно.

Тестування відбувається за рахунок спостереження за роботою продукту та аналізу даних результатів роботи продукту. Воно проходить в штучно створених ситуаціях і на штучно створених наборах даних, наближених до реальних.

В першу чергу, тестування проводиться для знаходження помилок в логіці роботи програмного забезпечення та для перевірки працездатності програмного забезпечення.

Під час розробки методу формування рекомендацій рекламних оголошень тестування проводилось на кожному етапі написання коду окремих модульних частин системи.

Також, проводилося тестування для перевірки спроможності працювати з великими обсягами даних та швидко їх обробляти.

За допомогою проведеного тестування забезпечено стабільну і правильну роботу системи та методу формування рекомендацій.

Одним із основних кінцевих способів тестування, була перевірка показників ефективності роботи методу відносно інших методів.

4.2. Оцінка роботи методу

Для проведення експерименту 1 було використано спеціально створений тестовий набір даних.

Цей набір даних включає в себе інформацію про рекламні оголошення, а також статистичні дані по рекламним оголошенням, які включають в себе інформацію про користувачів, які переглядали рекламні оголошення та їх оцінки.

Тестовий набір створених даних, містить в собі інформацію про 10000 різних рекламних оголошень та оцінки 50000 користувачів.

Для проведення експерименту обмежимося використанням демографічних даних про користувача, загальною інформацією про рекламні оголошення та трьома головними інформаційними полями:

- унікальний ідентифікатор рекламного оголошення;
- унікальний цифровий відбиток користувача, а точніше пристрою користувача;
- оцінка рекламного оголошення користувачем.

Демографічні дані включають в себе вік, стать та країну користувача.

Інформацію про головні інформаційні поля можна представити у вигляді наступної таблиці:

Таблиця 9

Вигляд головних інформаційних полів тестових даних

Унікальний ідентифікатор рекламного оголошення	Унікальний цифровий відбиток користувача	Оцінка рекламного оголошення
123	1	0
25	234	1
71	289	0
1289	4901	0
13	10001	0
...

Оцінки користувачів представлені у вигляді цифр 0 та 1, так як елементами системи є рекламні оголошення. Оцінка 0 означає що користувач не перейшов за посилання реклами, а тому вона його не зацікавила., Відповідно значення 1 означає, що користувач перейшов за посилання рекламного оголошення, тому вважається, що рекламне оголошення зацікавило його або відповідає його інтересам.

Наступний крок – це поділити експериментальні дані на дві частини у співвідношенні 25%до 75%:

- 25% – тестові дані;
- 75% – дані для обчислення.

Тобто, потрібно виконати прогноз оцінок елементів системи користувачами на основі даних для обчислення. Після цього потрібно порівняти оцінки, що були отримані, з тестовими даними. Це порівняння дасть нам розуміння про ефективність роботи розробленого модифікованого методу, а також покаже релевантність прогнозованих оцінок елементів системи.

Для того, щоб оцінити ефективність роботи алгоритмів будемо використовувати такі характеристики релевантності [37]:

- точність;
- повнота;
- F-міра.

Ці характеристики обчислюємо за такими формулами:

$$P = \frac{t_p}{t_p + f_p}, \quad (20)$$

$$R = \frac{t_p}{t_p + f_n}, \quad (21)$$

$$F = 2 \cdot \frac{P \cdot R}{P + R}, \quad (22)$$

де P – точність, R – повнота, F – F-міра відповідно, t_p – істинно-позитивні спрацювання, f_p – хибно-позитивні спрацювання, f_n – хибно-негативні спрацювання.

Істинно-позитивними спрацюваннями вважаються оцінки, які є позитивними як в тестових даних, так і в фінальних результатах роботи алгоритму.

Хибно-позитивними спрацюваннями вважаються оцінки, які є негативними у тестових даних, але у фінальному результаті роботи алгоритму вони негативні.

Хибно-негативними спрацюваннями вважаються оцінки, які є позитивними у тестових даних, але у фінальному результаті роботи алгоритму вони негативні.

Ефективність роботи методів будемо обчислювати за допомогою сумування трьох характеристик релевантності (формула 23).

$$E = P + R + F, \quad (23)$$

Експеримент 1 проводиться за допомогою використання різних методів формування рекомендацій:

- колаборативна фільтрація (CF);
- фільтрація по змісту (CB);
- демографічна фільтрація (DF);
- CB+CF;
- CF+DF;
- DF+CB;
- розроблений модифікований підхід (CF+CB+DF).

Результати експерименту 1 відображені у таблиці 10, а також на рис. 10.

Порівняння результатів роботи методів

Алгоритм	Точність	Повнота	F-міра	Ефективність
CF	0.611	0.53	0.575	1.716
CB	0.712	0.69	0.794	2.096
DF	0.65	0.63	0.64	1.92
CB+CF	0.637	0.643	0.641	1.921
CF+DF	0.601	0.607	0.651	1.859
DF+CB	0.65	0.658	0.655	1.963
Модифікований алгоритм	0.728	0.697	0.708	2.133

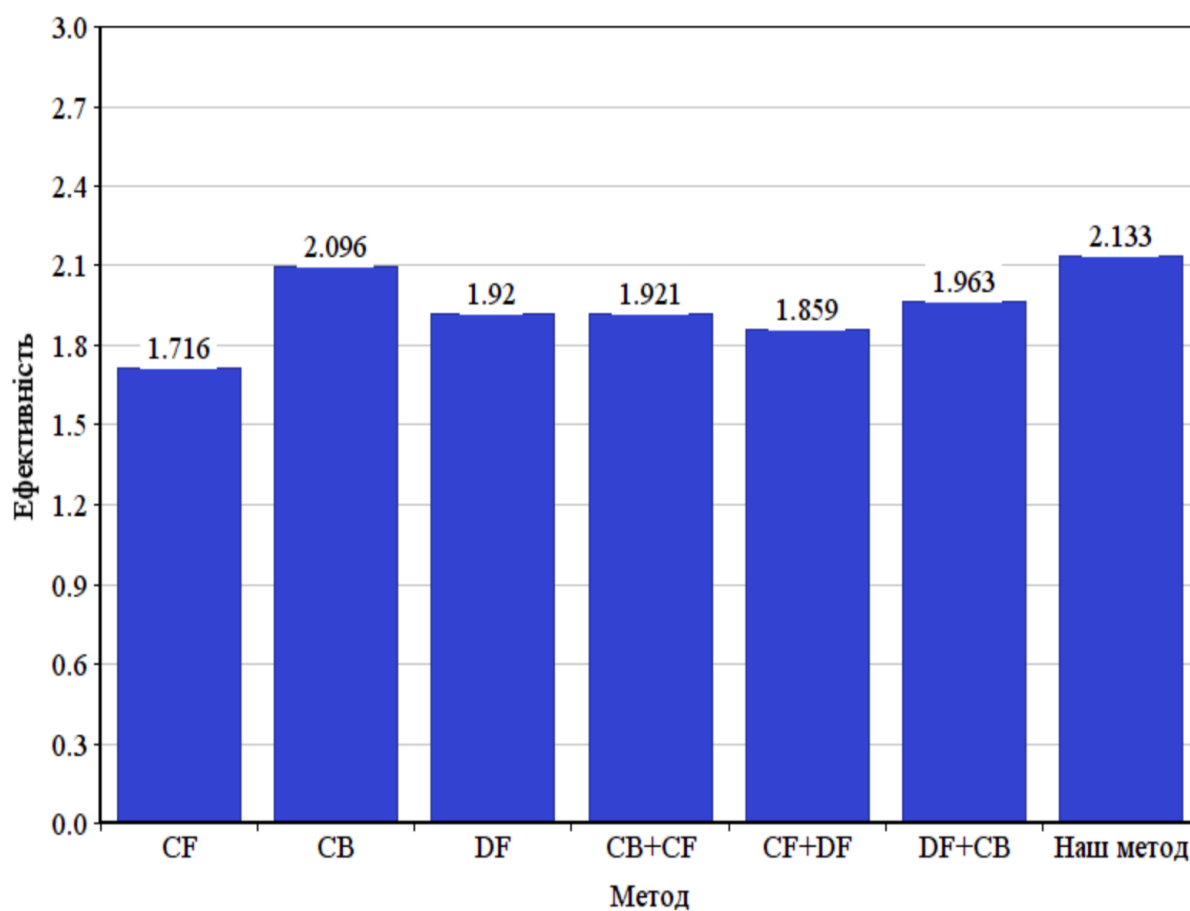


Рис. 10. Результати експерименту 1

Проведемо ще один експеримент 2.

Візьмемо такий самий набір даних як для експерименту 1, але цього разу не будемо ділити дані у відношенні 25% до 75%, а відберемо лише дані по певних 1000 користувачів, так щоб в даних для обчислення не залишилося інформації про цих користувачів та їх вподобання.

На основі даних про інших користувачів, а саме даних для обчислення, сформуємо список рекомендацій рекламних оголошення для нових користувачів з відібраних тестових даних.

За допомогою цього експерименту ми перевіримо роботу розробленого методу при виникненні проблеми холодного старту.

Будемо використовувати такі самі характеристики релевантності, ефективності, а також різні методи, як і для експерименту 1.

Результати експерименту 2 відображені у таблиці 11, а також на рис. 11.

Таблиця 11

Порівняння результатів роботи методів

Алгоритм	Точність	Повнота	F-міра	Ефективність
CF	0.23	0.25	0.24	0.72
CB	0.19	0.17	0.18	0.54
DF	0.61	0.62	0.615	1.845
CB+CF	0.2	0.21	0.205	0.615
CF+DF	0.47	0.45	0.46	1.38
DF+CB	0.45	0.46	0.455	1.315
Модифікований алгоритм	0.64	0.62	0.63	1.89

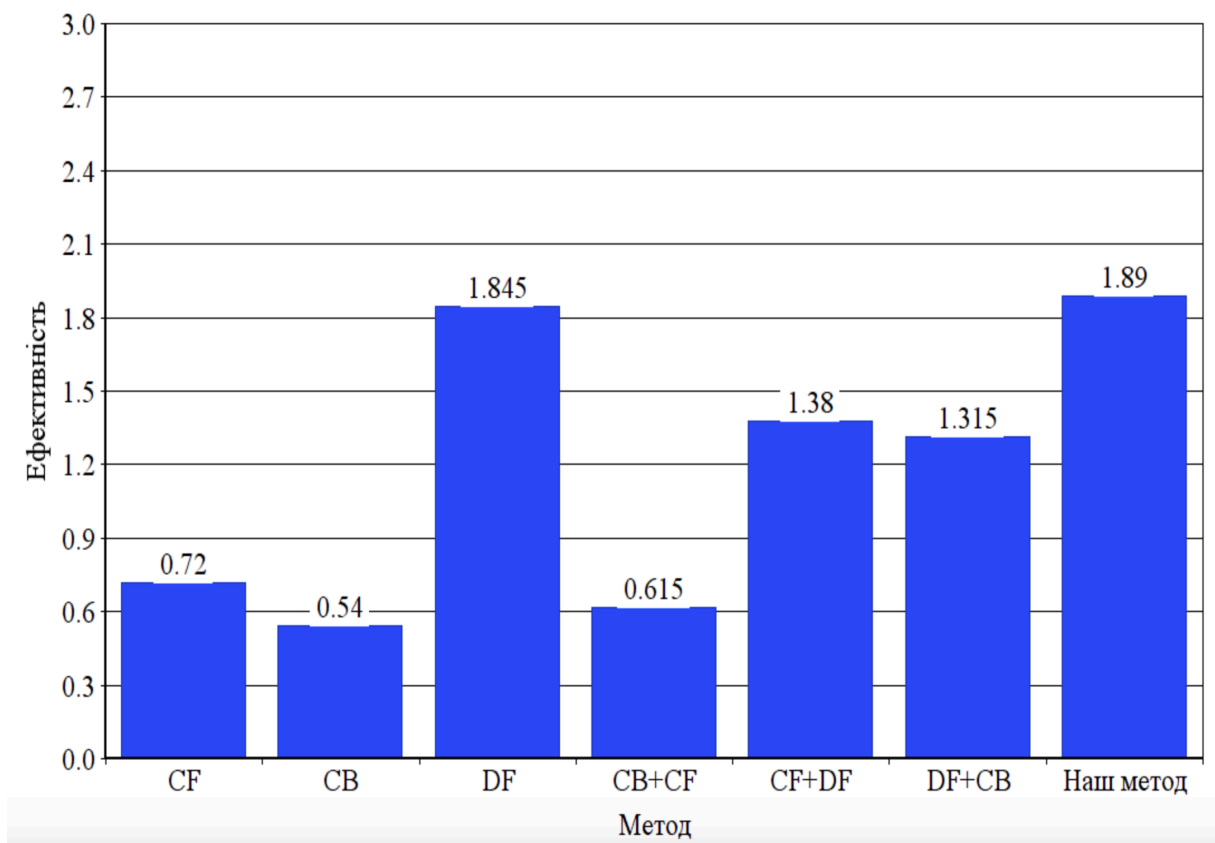


Рис. 11. Результати експерименту 2

Як можна побачити з результатів проведених експериментів, розроблений модифікований підхід дає досить точні прогнози у порівнянні з іншими підходами формування прогнозів, а саме забезпечується точність – 0.728, повнота – 0.697, F-міра – 0.708, ефективність – 2.133 для експерименту 1, та точність – 0.64, повнота – 0.62, F-міра – 0.63, ефективність – 1.89 для експерименту 2.

Чим більшим є кожен показник, а саме точність, повнота та F-міра, тим вища ефективність відповідного методу. У нашому випадку ефективність роботи методу була оцінена за допомогою додавання цих трьох показників релевантності рекомендацій.

Беручи за основу отримані в наслідок експерименту дані, можемо зробити висновок, що колаборативна фільтрація робить найбільший вплив на фінальний результат у експерименті 1, так як є в наявності велика кількість статистичних даних про оцінки користувачів системи, що

відповідно збільшує показник коефіцієнта γ та зменшує інші два вагових коефіцієнта α і β відповідно.

Але варто зазначити, що у разі виникнення проблеми холодного старту, як в експерименті 2, демографічна фільтрація вийде на перший план і зменшить показник коефіцієнта γ , що відповідно зменшить вплив отриманої оцінки методом колаборативної фільтрації на фінальний результат, тому що використання методу колаборативної фільтрації не буде таким ефективним, як демографічна фільтрація.

Розроблений модифікований підхід дозволяє формувати рекомендації рекламних оголошень на основі аналізу вподобань користувача.

Також, у разі виникнення проблеми холодного старту, запропонований метод використовує формулу, що об'єднує в собі параметри, котрі приймають різні значення в залежності від стану користувача з точки зору кількості наявних в системі оцінок.

Тобто параметри α , β і γ обчислюються динамічно, це робить розроблений модифікований метод більш гнучким та практичним у порівнянні з іншими існуючими методами.

4.3. Висновки до розділу

В цьому розділі було описано створені набори даних, які використовувалися для проведення експериментів. Описано характеристики релевантності, які використовувалися для оцінки ефективності роботи існуючих методів та розробленого метода формування рекомендацій.

Також у даному розділі було описано проведені експерименти, які продемонстрували ефективність розробленого модифікованого гібридного методу формування рекомендацій у порівнянні з іншими існуючими методами. Результати роботи розробленого модифікованого гібридного

методу виявилися трошки кращими за інші існуючі методи, з якими проводилося порівняння.

До подальших вдосконалень розробленого програмного методу формування рекомендацій:

- робота над мінімізацією часових затрат на обробку даних;
- збільшення показника точності прогнозування за рахунок врахування додаткових характеристик.

5. ПОБУДОВА БІЗНЕС-МОДЕЛІ

5.1. Опис бізнес-процесів

На сьогоднішній день, існує величезна кількість різноманітного контенту в мережі Інтернет. Це призводить до того, що звичайним користувачам дуже важко знайти саме ту інформацію, яка їм необхідна та цікавить їх. Іноді буває важко зробити вибір, навіть коли ти знаєш, що саме шукаєш. У таких ситуаціях на допомогу користувачам приходять рекомендаційні системи. Іноді навіть здається, що ця нежива система, яка працює за допомогою різних алгоритмів, краще знає, що хочуть користувачі, ніж самі користувачі.

На сьогоднішній день майже усі великі соціальні мережі, інтернет магазини, форуми, різноманітні сервіси, наприклад музикальні та відео сервіси, містять у собі рекомендаційну систему. В якості елемента рекомендацій може виступати фільм, книга, музика, реклама та інше. Рекомендаційні системи беруть свій початок ще у 90х роках, але популярності почали набирати лише нещодавно. Рекомендаційні системи зробили справжній прорив за останні роки, що вивело споживання контенту користувачами на новий рівень, також це зробило пошук інформації істотно легшим і зручнішим для користувачів.

5.1.1. Опис процесу діяльності

Кожен день людина має робити певний вибір. Рекомендаційні системи були створені саме для того, щоб допомагати користувачам робити вибір, не важливо чи це книга для читання, чи це комедійний фільм для перегляду з друзями, або ж курс по підвищенню фінансової усвідомленості, адже дуже часто користувачам не вистачає досвіду для цього свідомого вибору.

Для того, щоб надавати індивідуальні рекомендації користувачам, система збирає та аналізує інформацію про інших користувачів, а також

інформацію про елементи системи. Також, іноді системи використовують не тільки дані про вподобання користувачів, або данні про елементи системи, а неявні критерії, наприклад такі, як національність користувача, його вік, стать, місце знаходження та інші. Аналіз цієї різноманітної інформації набагато підвищує ефективність роботи рекомендаційної системи.

Якщо порівнювати між собою пошукові та рекомендаційні системи, то рекомендаційна система не працює з точно сформульованими запитами. В загальному випадку, говорячи про рекомендаційні системи, система пропонує користувачу надати оцінки декільком елементам системи, це робиться для того, щоб уникнути проблеми холодного старту, коли системі нічого невідомо про вподобання користувача, та на основі наданих користувачем оцінок, система може зробити прогноз вподобань користувача у майбутньому. За таким принципом працюють рекомендаційні систему у загальному випадку, тому що робота рекомендаційної системи повністю залежать від способу її реалізації. При роботі з рекомендаційною системою, ми завжди маємо справу з користувачами та множиною елементів системи, з яких користувач повинен обирати, варто зазначити, що про деякі елементи користувач може не здогадуватися.

Отже, робота рекомендаційної системи полягає в роботі алгоритму фільтрації існуючих вподобань користувачів, для того щоб сформувати рекомендації схожі на вже вподобані елементи, які вважаються найбільш релевантними для конкретного користувача системи.

Іноді, робота певного сервісу в повній мірі залежить від роботи рекомендаційної системи. Для прикладу можемо розглянути сервіси для прослуховування музики. Наприклад візьмемо музичний сервіс Spotify. Цей сервіс містить в собі близько 60 років аудіо потоку. Очевидно, що жоден з користувачів сервісу не буде слухати всі пісні, тому пошук нових пісень користувачем зазвичай ґрунтується на порадах друзів. На даному

етапі, задачею рекомендаційної системи є генерування персональних та релевантних рекомендацій для користувача, щодо нових пісень, на основі аналізу вподобань, історії поведінки користувача та пісень які він прослуховував в минулому.

Найбільш широкого застосування рекомендаційні системи знайшли у електронній комерції. Якщо розглядати рекомендаційні системи суто з точки зору бізнесу, то їх головною цілю є збільшення аудиторії зацікавлених користувачів, адже чим більш релевантний та цікавий контент пропонується користувачам, тим більше шанс, що користувачі будуть далі користуватися цим сервісом, а також запропонують його своїм знайомим та друзям. Також, варто зазначити, що завдяки рекомендаційним системам, зростають показники кількості проданих товарі, об'єм продажів, прибуток сервіс, час перебування користувача на сайті та інші.

Але рекомендаційні системи використовуються не лише для того, щоб рекомендувати користувачу певний товар або контент. Деякі інтернет-магазини застосовують рекомендаційні системи для того, щоб підвищити ефективність роботи акцій. В такому випадку, система робить прогноз зацікавленості кожного користувача в певному товарі або послугі, і на основі цих прогнозів, всім зацікавленим користувачам пропонуються акційні пропозиції або скидки на товари.

Компанії, які зазвичай використовують рекомендаційні системи, можна поділити на 2 основні типи:

- Компанії, зацікавлені в підвищенні продажів товарів та послуг. Існують користувачі, які купують товар, або оцінюють їх, а отже, необхідно надавати користувачам рекомендації, які будуть вважатися найбільш релевантними для них, це буде призводити до підвищення шансів на те, що користувачі будуть купувати товари, які задовольняють їх інтересам. До такого типу компаній можна віднести такі великі та відомі компанії, як Netflix, Amazon та інші.

- Компанії, які отримують головний прибуток від реклами. Тобто, реклама розміщена на сайті сервісу. При цьому, звичайно, потрібно показувати найбільш релевантні для користувача рекламні оголошення, це збільшить шанс на те, що користувач перейде за рекламним посиланням і сайт на якому розміщена реклама, отримає з цього прибуток.

Рекомендаційні системи мають 2 основні рівні на яких вона зазвичай функціонує:

- Глобальний рівень, це вподобання користувачів, які змінюються з великими проміжками часу, наприклад музичні вподобання, або вподобання до жанрів фільмів, а також інформація про самого користувача, така як вік, стать, національність та інші.
- Рівень короткочасних трендів, це короткочасні вподобання, які мають великий вплив, але на невеликому проміжку часу, наприклад сезонні події пов'язані з новим роком або іншими святами.

Розглянемо найпопулярніші рекомендаційні системи:

Amazon – один з лідерів в області рекомендаційних систем та першопроходець. Сервіс формує рекомендації, щодо товарів, основуючись на аналізі поведінки користувача, а саме історії придбаних товарів, оцінених товарів та переглянутих товарів.

Netflix – найбільшою популярністю користується на Заході, у нас же він тільки починає набирати оберти, але Netflix, це компанія, яка зараз найбільш стрімко розвивається в області рекомендаційних систем. На це дуже сильно вплинуло оголошення компанії про дуже велику винагороду, людині, яка зможе покращити якість та ефективність роботи алгоритму формування рекомендацій на 10%.

Google – не зважаючи на те, що Google це в основному пошукова система, вони також застосовуються алгоритми для формування рекомендацій. За допомогою рекомендаційної системи, вони роблять

прогноз, щоб дізнатися, наскільки певний сайт релевантний до певного пошукового запиту. Також варто зазначити, що майже у всіх таких великих пошукових систем існують інші проекти або сервіси, що представляють собою рекомендаційні системи, наприклад Google Music.

5.1.2. Актори і функції системи

За допомогою діаграми варіантів використання, покажемо акторів та функції системи, рис. 12.

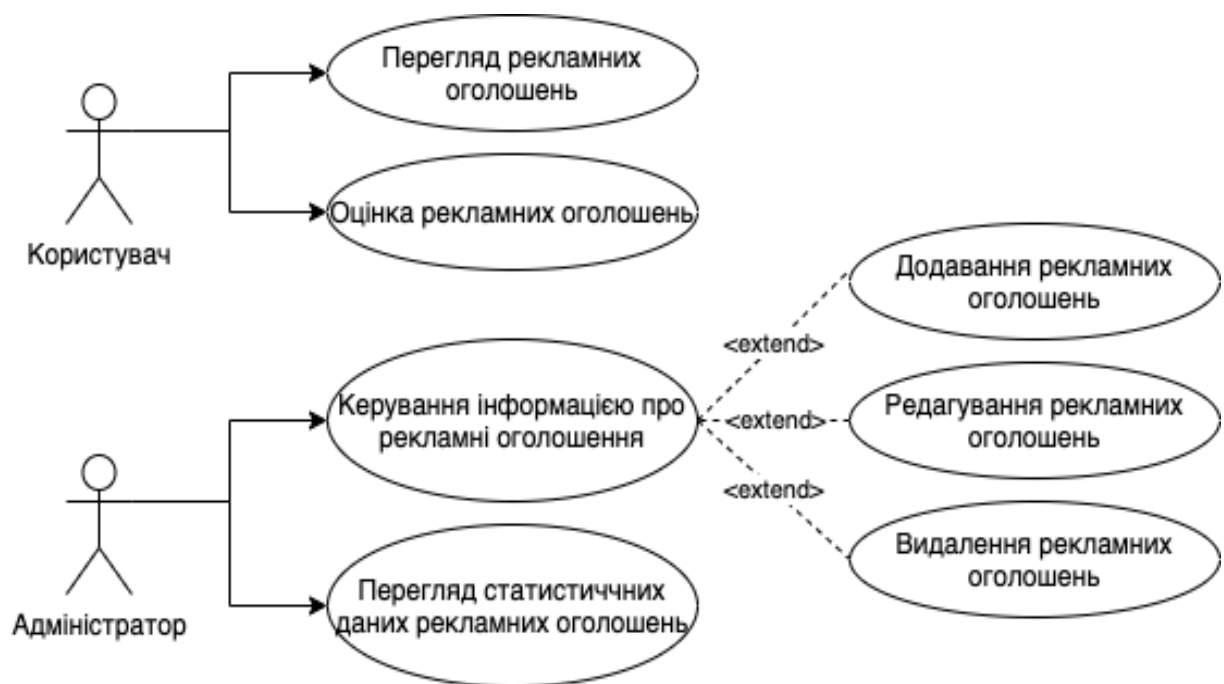


Рис. 12. Діаграма варіантів використання системи.

Як ми бачимо з діаграми, існує 2 типи акторів системи, це користувач та адміністратор. Кожен з акторів взаємодіє з системою таким чином: адміністратор може редагувати, додавати, видаляти інформацію про рекламні оголошення, а також переглядати статистичні дані рекламних оголошень, користувач може переглядати та оцінювати рекламні оголошення.

Опишемо ці варіанти використання більш детально.

Перегляд рекламних оголошень – користувач має можливість переглядати рекламні оголошення, коли він потрапляє на сайт, на якому розміщена реклама.

Оцінка рекламних оголошень – користувач має можливість оцінити рекламне оголошення. Так як ми маємо справу з рекламними оголошеннями, то оцінкою користувача вважається перехід користувача за посиланням рекламного оголошення, а саме клік по рекламі.

Керування інформацією про рекламні оголошення – у нашому випадку адміністратор займається роботою з рекламними оголошеннями. Він може додавати, редагувати та видаляти рекламні оголошення. Варто зазначити, що на реальному рекламному сервісі, це роблять рекламодавці, які повинні пройти попередню реєстрацію у сервісі, але ми розглядаємо спрощений варіант системи, в якій елементами системи виступають рекламні оголошення.

Перегляд статистичних даних рекламних оголошень – адміністратор має можливість переглядати статистичні дані рекламних оголошень, вони включають у себе перегляди рекламних оголошень та переходи за посиланням рекламного оголошення користувачами.

5.1.3. Бізнес-процеси системи

Основними процесами діяльності системи є ведення рекламних оголошень та перегляд рекламних оголошень, які розміщені на сайтах, користувачами.

Для початку роботи рекомендаційної системи, адміністратору потрібно заповнити систему елементами, а саме рекламними оголошеннями. Варто зазначити, що ми розглядаємо спрощену модель роботи системи для розробки методу формування рекомендацій, в реальній ситуації рекламні оголошення створюються рекламодавцями, які зареєстровані на рекламному сервісі або платформі. Після того, як у системі з'являться рекламні оголошення, можливо буде формувати

рекомендації, щодо них, для користувачів, які бачать рекламні оголошення на різних сайтах.

При відвідуванні сайтів, на яких розміщені рекламні блоки, робиться запит на отримання рекламного оголошення, яке буде показано користувачу. Відбувається формування персональних рекомендацій на основі аналізу вподобань користувача у минулому. Кожен користувач має свій унікальний фінгерпринт, за допомогою якого можливе розпізнавання користувачів.

Також, в якості параметрів, які впливають на формування списку рекомендацій, можуть бути демографічні дані про користувача, такі як вік, стать, національність, місце знаходження, так і дані про елементи системи, такі як категорія реклами, заголовок, тип та інші.

5.2. Опис постановки задачі

Головне призначення системи – це надання користувачам персональних рекомендацій, щодо найбільш релевантних для них елементів системи, а саме рекламних оголошень, які будуть ним найбільш цікавими.

Ціль створення системи – максимальне задоволення потреб та вподобань користувачів за допомогою формування персональних рекомендацій, на основі аналізу інформації про користувача та елементи системи, а також вподобань інших користувачів системи.

Для реалізації вищезазначених цілей, система має вирішувати такі задачі:

- можливість роботи з рекламними оголошеннями, що включає в себе додавання, редагування та видалення рекламних оголошень системи;
- збір та аналіз статистичних даних про вподобання користувачів системи;

- формування списку персональних рекомендацій для кожного користувача системи;
- вибір та надання найбільш релевантних елементів системи для кожного користувача системи.

5.3. Огляд аналогів системи

На сьогоднішній день, існує дуже велика кількість різноманітних рекомендаційних систем. Умовно їх можна поділити на групи, основуючись на елементах системи, щодо яких надаються рекомендації користувачам. Були сформовані такі групи:

- Аудіо контент;
- Відео контент;
- Товари;
- Рекламні оголошення;
- Новини;
- Інше.

Рекомендаційні системи, фокусуються лише на певному виді контенту, тому вони відрізняються між собою, а також відрізняються реалізацією алгоритмів для роботи системи.

Розглянемо сервіси, які відносяться до кожної з сформованих вище груп.

Spotify – це один з найпопулярніших сервісів у світі, який використовується для прослуховування аудіо контенту, а саме музики. Цей сервіс містить в собі одну з найбільших аудіо бібліотек, також він представляє користувачам можливість безкоштовно прослуховувати музику. Цей сервіс був одним із перших, хто надав таку можливість. Сервіс доступний на території Америки, майже усій території Європи, у певних країнах Азії, а також в Україні. Сервіс включає в себе рекомендаційну систему, яка оперує музичними композиціями, як елементами системи для формування рекомендацій.

YouTube – сервіс, основний функціонал якого є зберігання та показ відео. За допомогою цього сервісу, користувачі можуть завантажувати свої відео, переглядати відео інших користувачів, давати оцінки, залишати коментарі до відео, а також ділитися з іншими користувачами. Завдяки своїй простоті та зручності у використанні сервісу, він став найпопулярнішим відеохостингом у світі, а також, сервіс займає друге місце за кількість відвідувачів. YouTube надає рекомендації 2х типів: глобальні або світові рекомендації, тобто відео записи, які вважаються найпопулярнішими на даний момент, персональні рекомендації, тобто відео записи, які система пропонує користувачу, на основі аналіз історії, поведінки та вподобань користувача.

Amazon – компанія, яка є найбільшою у світі в сфері інтернет торгівлі товарами та послугами. Також, це один із перших сервісів, який орієнтований на продаж реальних товарів масового попиту. Рекомендаційна система Amazon, формує рекомендації товарів основуючись на інформації про минулі покупки користувача, перегляди товарів, поставлені оцінки та залишені відгуки.

Google News – агрегатор новин від компанії Google. Формування рекомендацій відбувається на основі наявних даних про користувача системи, таких як вік, стать, національність, місце знаходження та на основі пошукових запитів зроблених користувачем у системі Google.

Якщо розглядати рекламні рекомендаційні системи, то однією з найбільших та найвідоміших є Google Ads, також продукт від компанії Google. Це сервіс контекстної, в основному, пошукової реклами, який надає рекламодавцям зручний інтерфейс та різноманітні інструменти для створення ефективних рекламних кампаній.

На даний момент існує багато великих та малих сервісів для рекламування, але більшість з них надає неякісні рекомендації користувачам, а деякі взагалі не використовують рекомендаційні системи, а лише беруть до уваги ціну реклами, тобто прибуток який буде отриманий

сервісом з цієї рекламної кампанії. Також, варто брати до уваги, що сфера рекомендаційних систем зараз дуже швидко розвивається. Тому можна зазначити, що розробка нового методу формування рекомендацій рекламних оголошень, на основі аналізу вподобань користувачів є доцільною.

5.4. Опис ідеї стартап-проекту

Основна ідея стартап-проекту полягає у наступному (табл. 12).

Таблиця 12

Опис ідеї стартап-проекту

Зміст ідеї	Напрямки застосування	Вигода для користувачів
Програмний метод формування рекомендацій рекламних оголошень на основі аналізу вподобань користувачів, який дозволить користувачам отримувати релевантні рекламні оголошення відповідно до їх інтересів	Застосування у рекламних сервісах або платформах, для підвищення якості рекламних оголошень, які отримують користувачі	Отримання рекламних рекомендацій відповідно до вподобань користувача, що буде задовольняти інтересам користувача і підвищувати зацікавленість у рекламованому товарі

5.5. Аналіз ринкових можливостей запуску стартап-проекту

Характеристика потенційних клієнтів стартап-проекту, описана в табл.13.

Таблиця 13

Характеристика потенційних клієнтів стартап-проекту

Потреба, що формує ринок	Цільова аудиторія	Відмінність у поведінці різних потенційних груп користувачів	Вимогу споживачів до товару
Необхідність отримання релевантних рекламних оголошень користувачами	Будь-які користувачі мережі інтернет	Різниця у поведінці за різними факторами: вік, стать, національність та інші	Релевантні рекомендації, щодо рекламних оголошень, які отримують користувачі

Основні фактори загроз для стартап-проекту описані в табл.14.

Таблиця 14

Фактори загроз

Фактор	Зміст загрози	Можлива реакція компанії
Зниження інтересу користувачів до рекламних оголошень	Зниження популярності предметної області – інтернет-реклами	Відсутня
Недостатня інформаційна база на початку існування проекту	Недостатня інформація про вподобання користувачів	Метод буде формувати рекомендації на основі демографічних даних
Недостатня популярність	Відсутність уваги до методу з точки зору клієнтів	Рекламна кампанія. Пошук зацікавлених клієнтів

Основні фактори можливостей стартап-проекту описані в табл. 15.

Таблиця 15

Фактори можливостей

Фактор	Зміст можливості	Можлива реакція компанії
Партнерство з рекламними платформами або сервісами	Збільшення кількості клієнтів.	Залучення партнерів.

Альтернативне ринкове провадження описано в табл. 16.

Таблиця 16

Альтернативне ринкове впровадження

Альтернатива ринкової поведінки	Ймовірність отримання ресурсів	Строки реалізації
Випуск MVP- версії в публічний доступ	Ресурси надані інвесторами	6 місяців
Створення власної рекламної платформи для використання методу формування рекомендацій	Ресурси надані інвесторами	2 роки

5.6. Висновки до розділу

У даному розділі було побудовано бізнес-модель запропонованого модифікованого методу формування рекомендацій, а саме:

- розглянуто бізнес-процеси;
- описано акторів та функції системи;
- проведено аналіз існуючих аналогів систем, які використовують методи формування рекомендацій;
- описано ідею стартап-проєкту;

- проведено аналіз потенційних клієнтів;
- розглянуто фактори можливостей та загроз;
- розглянуто альтернативне ринкове впровадження.

ВИСНОВКИ

У даній магістерській дисертації запропоновано модифікований гібридний метод формування рекомендацій рекламних оголошень на основі аналізу вподобань користувачів з метою підвищення ефективності роботи рекламних сервісів та платформ.

Проаналізовано існуючі методи формування рекомендацій, основні з них: колаборативна фільтрація, демографічна фільтрація, фільтрація на основі вмісту та гібридні методи фільтрації. Визначено основні переваги та недоліки кожного з методів.

За результатами проведеного аналізу було запропоновано модифікований гібридний метод формування рекомендацій, який ґрунтується на трьох методах фільтрації: колаборативному, демографічному та фільтрації на основі вмісту.

Проведено експериментальні дослідження, за результати яких була показана ефективність роботи запропонованого модифікованого методу на різних наборах даних. Запропонований модифікований метод показує себе трошки краще ніж інші існуючі методи з якими проводилося порівняння.

Описано засоби розробки програмного забезпечення, структуру бази даних системи за допомогою ER-діаграми, описано кожну таблицю та кожне поле таблиць бази даних.

Розроблено діаграму класів, діаграму послідовності роботи системи та діаграму взаємодії компонентів системи.

Розглянуто існуючі аналоги систем, які використовують методи формування рекомендацій.

Описано бізнес-модель, а також ідея стартап-проекту. Також, проведено аналіз ринкових можливостей, загроз та альтернативного ринкового провадження.

Основні положення і результати роботи були представлені та обговорювались на XIII науковій конференції магістрантів та аспірантів

«Прикладна математика та комп'ютинг» ПМК-2020, яка проходила у місті Київ з 18 по 20 листопада 2020р.

Практичне застосування результатів даної магістерської дисертації дозволить:

- користувачам отримувати персоналізовані рекомендації рекламних оголошень;
- рекламним платформам підвищити ефективність своїх рекламних оголошень.

СПИСОК ЛІТЕРАТУРНИХ ДЖЕРЕЛ

1. Архітектура програмного забезпечення – [Електронний ресурс] – Режим доступу: <https://bit.ly/2Lzp6NH>
2. База даних – [Електронний ресурс] – Режим доступу: <https://bit.ly/3nfLFPh>
3. Програмне забезпечення – [Електронний ресурс] – Режим доступу: <https://bit.ly/3gNJvEf>
4. Фреймворк – [Електронний ресурс] – Режим доступу: <https://bit.ly/37e3h8K>
5. СКБД – [Електронний ресурс] – Режим доступу: <https://bit.ly/3oImW6L>
6. Колаборативна фільтрація (CB) – [Електронний ресурс] – Режим доступу: <https://bit.ly/2W9nBSq>
7. CSS – [Електронний ресурс] – Режим доступу: <https://uk.wikipedia.org/wiki/CSS>
8. ER-модель – [Електронний ресурс] – Режим доступу: <https://bit.ly/3a06QRQ>
9. HTML – [Електронний ресурс] – Режим доступу: <https://uk.wikipedia.org/wiki/HTML>
10. MongoDB – [Електронний ресурс] – Режим доступу: <https://uk.wikipedia.org/wiki/MongoDB>
11. Node.js – [Електронний ресурс] – Режим доступу: <https://uk.wikipedia.org/wiki/Node.js>
12. ORM – [Електронний ресурс] – Режим доступу: <https://ru.wikipedia.org/wiki/ORM>
13. IDEF0 – [Електронний ресурс] – Режим доступу: <https://uk.wikipedia.org/wiki/IDEF0>
14. TF-IDF міра – [Електронний ресурс] – Режим доступу: <https://uk.wikipedia.org/wiki/TF-IDF>
15. JavaScript – [Електронний ресурс] – Режим доступу: <https://uk.wikipedia.org/wiki/JavaScript>

16. JSON – [Электронный ресурс] – Режим доступа:
<https://uk.wikipedia.org/wiki/JSON>
17. J. Bobadilla, F. Ortega, A. Hernando, and A. Guti´errez, “Recommender systems survey,” *Knowledge-Based Systems*, vol. 46, 2013 – P. 109-132.
18. В.В. Куриленко Разработка веб-приложения музыкального аудиостриминга с рекомендательной системой // НИ ТГУ – 2017 – P. 7-15.
19. G. Salton Automatic Text Processing // Addison-Wesley. – 1989 – 530 pages.
20. Daniar Asanov (n.d.) Algorithms and Methods in Recommender Systems, Berlin, Germany: Berlin Institute of Technology – 2011 – P. 1-7.
21. Jones, M. T. (2013, December 12). Recommender systems.Introduction to approaches and algorithms. Retrieved November 25, 2017, from <https://www.ibm.com/developerworks/library/os-recommender1/>
22. Jerold Angelus Grundy Newbrain // Duct Publishing – 2012 – 120 pages.
23. D. Goldberg Using Collaborative Filtering to Weave an Information Tapestry / D. Goldberg, D. Nichols, B.M. Oki, D. Terry // *Comm. ACM.* – 1992. – Vol. 35, No12. – P. 61-70.
24. Gleb Beliakov, Tomasa Calvo and Simon James. Aggregation of preferences in recommender systems / Gleb Beliakov and Simon James // School of Information Technology, Deakin University – 2008 – P. 705-735.
25. J. Basilico and T. Hofmann. Unifying collaborative and content-based filtering. In *Proceedings of the 21th International Conference on Machine Learning*, 2004 – P. 9-16
26. C.C. Aggarwal Horting Hatches an Egg: A New Graph-Theoretic Approach to Collaborative Filtering / C.C. Aggarwal, J.L. Wolf, K-L. Wu, P.S. Yu // *Proc. Fifth ACM SIGKDD Int’l Conf. Knowledge Discovery and Data Mining.* – 1999 – P. 201-212.

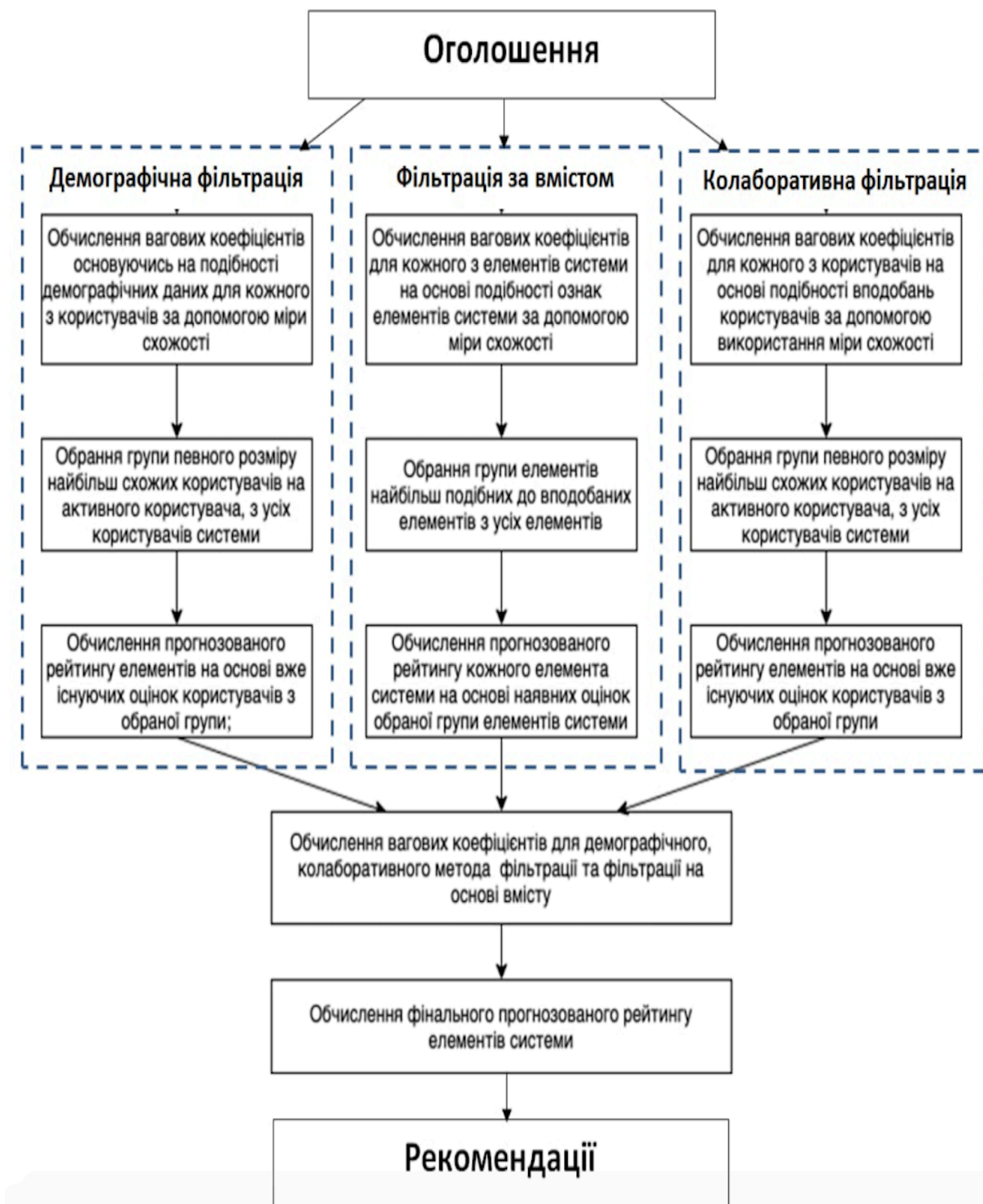
27. P. Resnick GroupLens: An Open Architecture for Collaborative Filtering of Netnews / P. Resnick, N. Iakovou, M. Sushak, P. Bergstrom, J. Riedl // Proc. 1994 Computer Supported Cooperative Work Conf. – 1994 – P. 175-186.
28. U. Shardanand Social Information Filtering: Algorithms for Automating “Word of Mouth” / U. Shardanand, P. Maes // Proc. Conf. Human Factors in Computing Systems. – 1995 – P. 210-217.
29. J.S. Breese Empirical Analysis of Predictive Algorithms for Collaborative Filtering / J.S. Breese, D. Heckerman, C. Kadie // Proc. 14th Conf. Uncertainty in Artificial Intelligence. – 1998 – P. 43-52.
30. B. Sarwar Item-Based Collaborative Filtering Recommendation Algorithms / B. Sarwar, G. Karypis, J. Konstan, J. Riedl // Proc. 10th Int’l WWW Conf. – 2001 – P. 285- 295.
31. J. Delgado Memory-Based Weighted-Majority Prediction for Recommender Systems / J. Delgado, N. Ishii // Proc. ACM SIGIR ’99 Workshop Recommender Systems: Algorithms and Evaluation. – 1999 – P. 186-198.
32. A. Nakamura Collaborative Filtering Using Weighted Majority Prediction Algorithms / A. Nakamura, N. Abe //Proc. 15th Int’l Conf. Machine Learning. – 1998 – P. 395-403.
33. I. Soboroff Combining Content and Collaboration in Text Filtering / I. Soboroff, C. Nicholas // Proc. Int’l Joint Conf. Artificial Intelligence Workshop: Machine Learning for Information Filtering. – 1999 – P. 86-91.
34. A.I. Schein Methods and Metrics for Cold-Start Recommendations / A.I. Schein, A. Popescul, L.H. Ungar, D.M. Pennock // Proc. 25th Ann. Int’l ACM SIGIR Conf. – 2002 – P. 253-260.
35. А. В. Заболеева-Зотова Латентный семантический анализ: новые решения в Internet / А. В. Заболеева-Зотова, А. Ю. Пастухов, П. В. Сердюков, Н. А. Козлова, С. А. Чернов // Информационные технологии. – 2001 – P. 67-82.

36. E. Aïmeur, G. Brassard, J. M. Fernandez, and F. S. M. Onana, Privacy-preserving demographic filtering, in Proceedings of the ACM symposium on Applied computing. New York, NY, USA: ACM, 2006 – P. 872–878.
37. Powers, David M. W. Evaluation: From Precision, Recall and F-Measure to ROC, Informedness, Markedness & Correlation // Journal of Machine Learning Technologies, 2011 – P. 37–63.

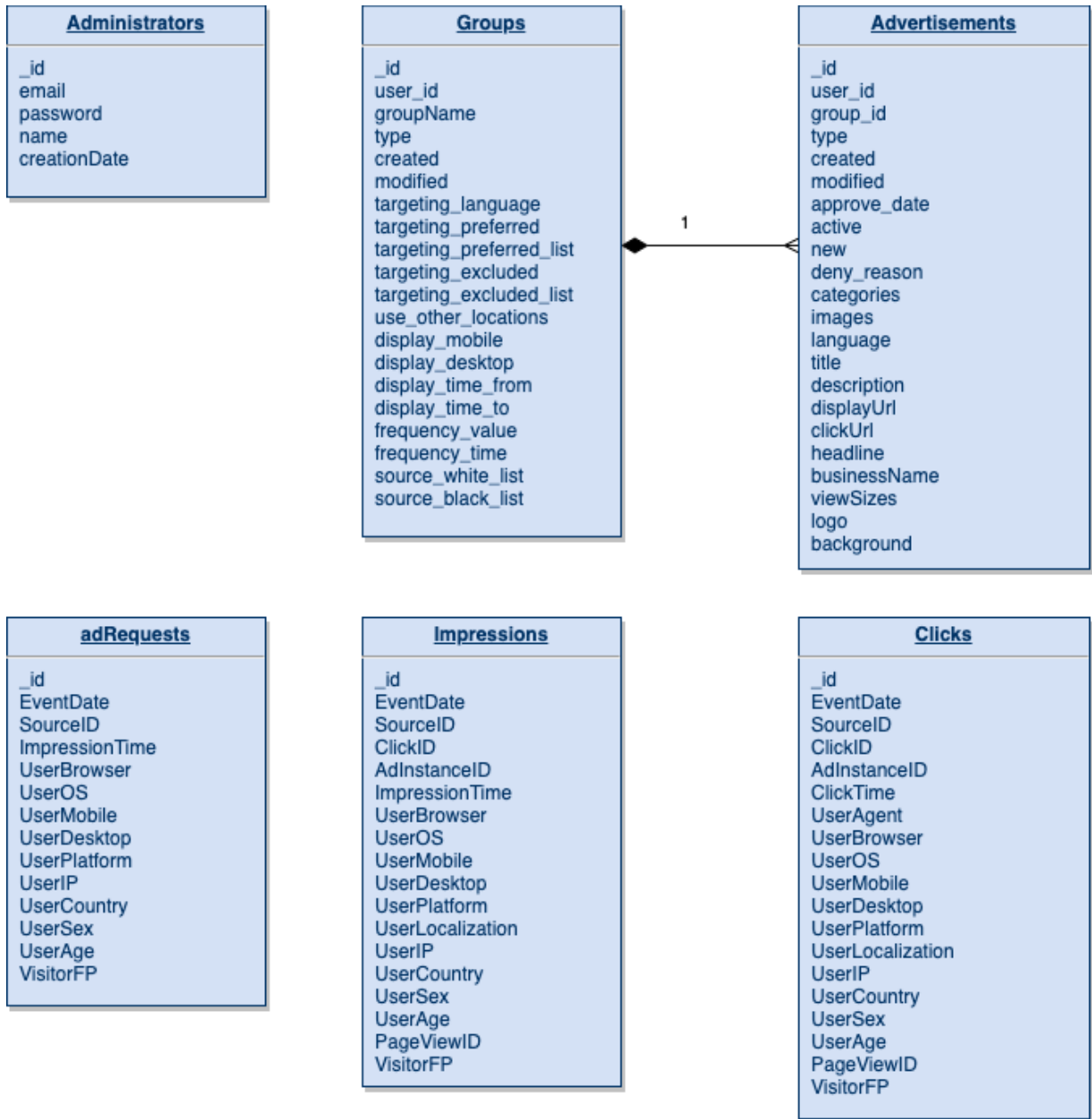
ДОДАТКИ

Додаток 1
Копії графічних матеріалів

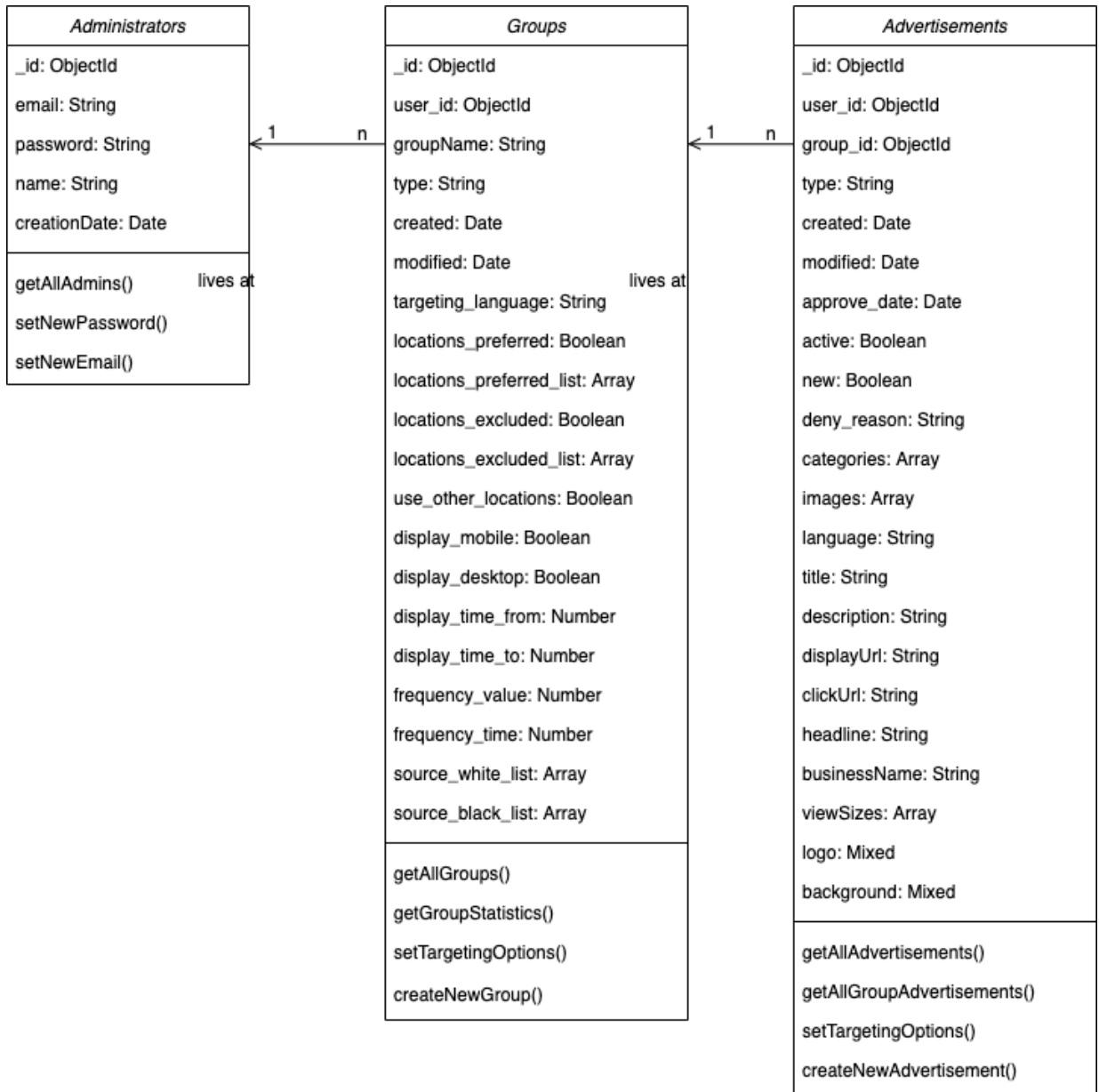
Схема алгоритму роботи модифікованого методу



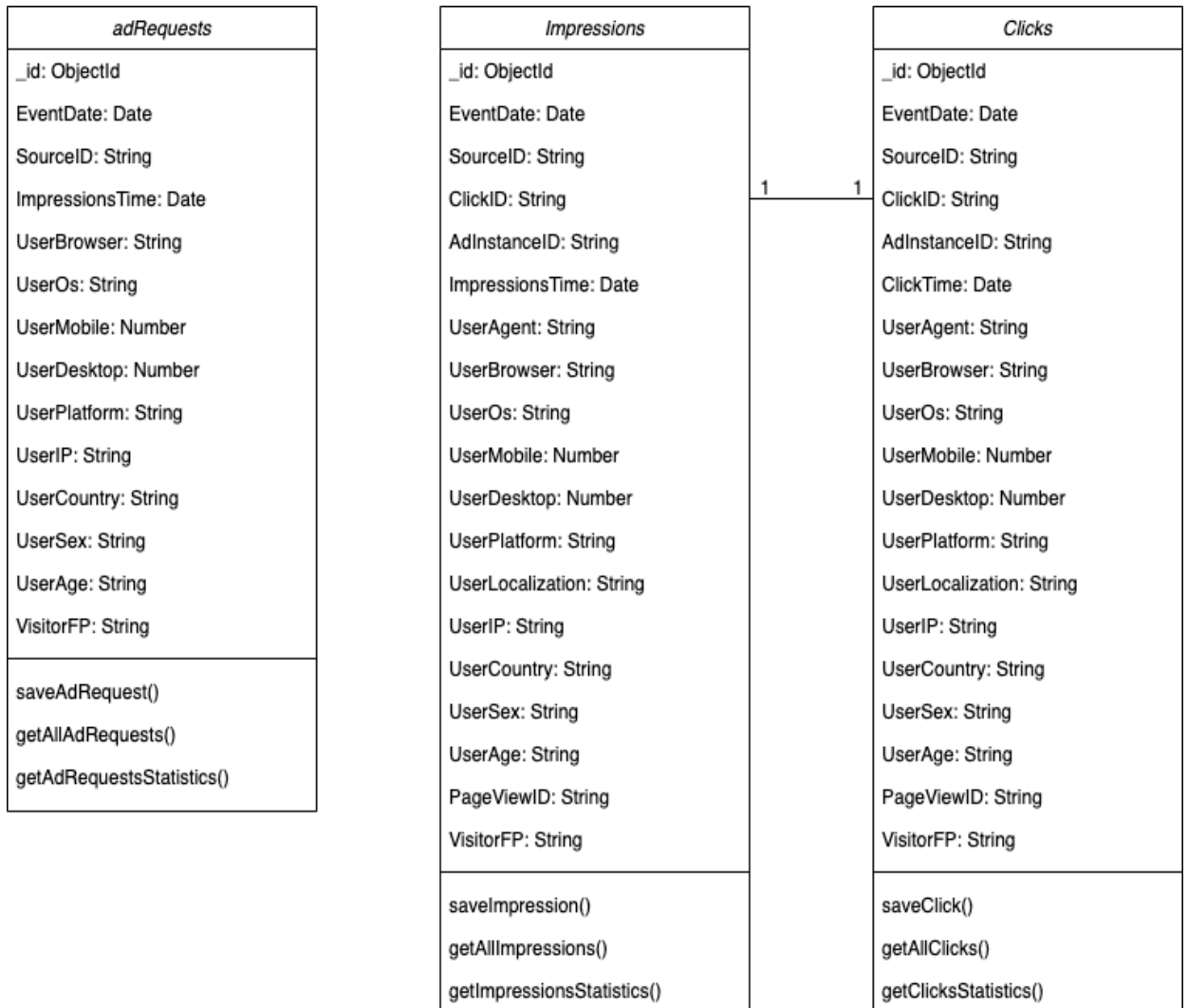
ER-діаграма бази даних



Діаграма класів. Частина 1



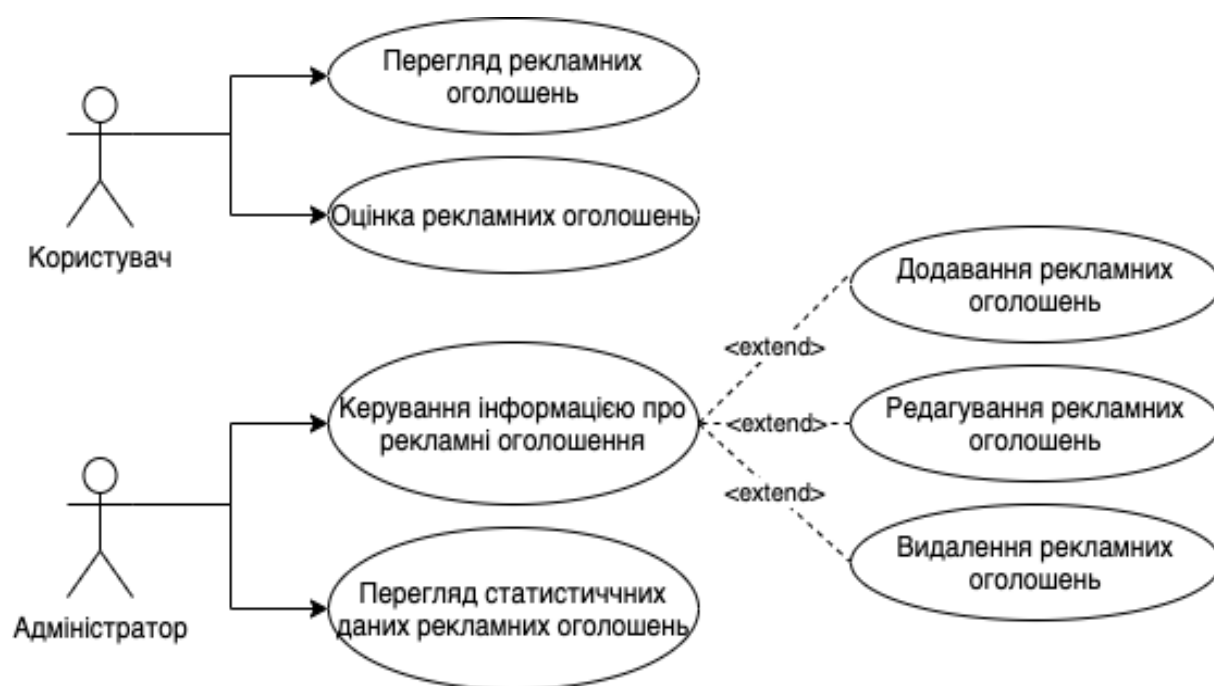
Діаграма класів. Частина 2



Діаграма послідовності



Діаграма варіантів використання системи



Додаток 2
Лістинг програми

Лістинг 1. Метод формування рекомендацій

```
const mongoose = require("mongoose");

const { Schema } = mongoose;
const advertSchema = new Schema(
  {
    user_id: { type: String, index: true },
    group_id: { type: String, index: true },
    campaign_id: { type: String, index: true },
    epom_id: { type: String, index: true },
    modified: { type: Date, index: true, default: Date.now },
    approved: { type: Boolean, index: true, default: false },
    approve_date: { type: Date },
    first_approve_date: { type: Date },
    cpa: { type: Boolean, index: true, default: false },
    active: { type: Boolean, index: true, default: false },
    new: { type: Boolean, index: true, default: true },
    deny_reason: { type: String, index: false, default:
"MANUALLY_STOPPED" },
    sensitive_content: { type: [String], default: [] },
    categories: { type: [String], default: ["default"] },
    ad_type: { type: String, index: true, default: "text" },
    img_type: { type: String, index: true },
    created: { type: Date, index: true, default: Date.now },
    decided: { type: Date, index: true },
    details: {
      images: { type: mongoose.Schema.Types.Mixed },
      language: String,
      title: String,
      description1: String,
      description2: String,
      displayurl: String,
      clickurl: String,
      bid: { type: Number, index: true, default: 0 },
      runtimePrediction: { type: Number, default: 0 },
      headline: String,
      businessName: String,
      logo: { type: mongoose.Schema.Types.Mixed },
      background: { type: mongoose.Schema.Types.Mixed },
      html5Directory: { type: String },
      viewSizes: { type: Array },
    },
    videoFileAccessUrl: { type: String, index: true },
  },
  { collection: "adverts" }
);

advertSchema.statics.MANUALLY_STOPPED = "MANUALLY_STOPPED";
advertSchema.statics.LOW_BALANCE = "CAMPAIGN_LOW_BALANCE";
advertSchema.statics.LIMIT_REACHED_PAUSE = "LIMIT_REACHED_PAUSE";

module.exports.Advertisement = mongoose.model("Advertisement",
advertSchema);

const mongoose = require("mongoose");

const { Schema } = mongoose;
const groupSchema = new Schema(
  {
    user_id: {
      type: String,
      index: true,
```

```

    },
    campaign_id: {
      type: String,
      index: true,
    },
    groupName: {
      type: String,
      index: true,
    },
    type: {
      type: String,
      default: "",
    },
    createdAt: {
      type: Date,
      default: Date.now,
    },
    modifiedAt: {
      type: Date,
      default: Date.now,
    },
    limit: {
      exists: {
        type: Boolean,
        index: true,
        default: false,
      },
      amount: {
        type: Number,
        index: true,
      },
      reached: {
        type: Boolean,
        index: true,
        default: false,
      },
      stop_time: {
        type: Date,
        index: true,
      },
    },
    targeting_options: {
      language: {
        type: String,
        default: "English",
        index: true,
      },
      locations_preferred: {
        type: Boolean,
        default: false,
      },
      locations_preferred_list: {
        type: [String],
        default: [],
      },
      locations_excluded: {
        type: Boolean,
        default: false,
      },
      locations_excluded_list: {
        type: [String],
        default: [],
      },
      buy_other: {

```

```

        type: Boolean,
        default: true,
    },
    buy_other_value: {
        type: Number,
        default: 60,
    },
    display_desktop: {
        type: Boolean,
        default: true,
    },
    display_mobile: {
        type: Boolean,
        default: true,
    },
    display_mobile_os: {
        android: {
            type: Boolean,
            default: true,
        },
        ios: {
            type: Boolean,
            default: true,
        },
        winmobile: {
            type: Boolean,
            default: true,
        },
        others: {
            type: Boolean,
            default: true,
        },
    },
    display_desktop_os: {
        windows: {
            type: Boolean,
            default: true,
        },
        linux: {
            type: Boolean,
            default: true,
        },
        macos: {
            type: Boolean,
            default: true,
        },
        others: {
            type: Boolean,
            default: true,
        },
    },
    ad_rerun: {
        type: Number,
        default: 72,
    },
    display_time_range: {
        from: {
            type: Number,
            default: 0,
        },
        to: {
            type: Number,
            default: 24,
        },
    },

```



```

    },
    frequency_capping: {
      type: Boolean,
      default: false,
    },
    frequency_value: {
      type: Number,
      default: 0,
    },
    frequency_time: {
      type: Number,
      default: 24,
    },
    frequency_target: {
      type: String,
      default: "advertisement",
    },
    blocked_sources: {
      type: [String],
      default: [],
    },
    white_list: {
      type: [String],
      default: [],
    },
    buy_other_forlist: {
      type: Boolean,
      default: false,
    },
    buy_other_value_forlist: {
      type: Number,
      default: 60,
    },
    blacklist_enable: {
      type: Boolean,
      default: false,
    },
    sources_custom_bids_list: {
      type: [
        {
          date: {
            type: Date,
          },
          source_id: {
            type: String,
          },
          bid: {
            type: Number,
            default: 0,
          },
        },
      ],
    },
    bid: {
      type: Number,
      index: true,
    },
    selected_package_id: {
      type: String,
      default: null,
    },
    isArchived: {
      type: Boolean,

```

```

        index: true,
        default: false,
    },
    isInternal: {
        type: Boolean,
        index: true,
        default: false,
    },
},
{ collection: "groups" }
);

module.exports.Group = mongoose.model("Group", groupSchema);

const mongoose = require("mongoose");

const { Schema } = mongoose;
const User = new Schema(
{
    full_name: { type: String },
    contact_info: {
        additional_email: { type: String },
        country: { type: String },
        timezone: { type: Number, index: true, default: 0 },
        contact_phone: { type: String },
        messengers: [
            {
                type: { type: String },
                url: { type: String },
                id: { type: String },
            },
        ],
    },
    advert_init_date: { type: Date, index: true },
    local: {
        email: { type: String, index: true, unique: true },
        password: String,
    },
    status: {
        active_publisher: { type: Boolean, index: true, default:
false },
        active: { type: Boolean, index: true, default: null },
        approved: { type: Boolean, index: true, default: false },
        approval_date: { type: Date },
        adminrules: Boolean,
        card_payment: {
            type: Boolean,
            default: true,
        },
    },
    lastLogin: { type: Date },
    lastAccess: { type: Date },
    registered: { type: Date, default: Date.now },
    advDate: { type: Date },
    pubDate: { type: Date },
    referrer: {
        id: { type: String },
        url: { type: String },
        limit: { type: Number, default: 20 },
    },
    balance: {
        btc_address: { type: String, index: true },
    },
    loginInfo: {

```

```

        ip: String,
        country: String,
        city: String,
        location: {
            latitude: Number,
            longitude: Number,
            time_zone: String,
            accuracy_radius: Number,
        },
    },
    lastAccessInfo: {
        ip: String,
        country: String,
        city: String,
        location: {
            latitude: Number,
            longitude: Number,
            time_zone: String,
            accuracy_radius: Number,
        },
    },
    registerInfo: {
        ip: String,
        country: String,
        city: String,
        location: {
            latitude: Number,
            longitude: Number,
            time_zone: String,
            accuracy_radius: Number,
        },
    },
    facebook: {
        id: { type: String, index: true },
        token: { type: String, index: true },
        email: { type: String, index: true },
        name: String,
    },
    emailSubscription: { type: Boolean, default: true },
    emailCustomiseMsg: { type: Boolean, default: true },
    emailSent: { type: Boolean, default: false },
    bonus: {
        type: { type: String },
        spent: { type: Boolean },
        quest_pass: { type: Boolean, default: false },
        quest_id: { type: String },
    },
    dormant_pub: {
        is_dormant: {
            type: Boolean,
            default: false,
        },
        page_views_48h: {
            type: Number,
            default: 0,
        },
        page_views_1w: {
            type: Number,
            default: 0,
        },
        impressions_48h: {
            type: Number,
            default: 0,
        },
    },

```

```

        impressions_1w: {
            type: Number,
            default: 0,
        },
        check_period: {
            type: Number,
            default: 24,
        },
        logs: [
            {
                date: { type: Date },
                reason: { type: String },
            },
        ],
    },
    dormant_adv: {
        is_dormant: {
            type: Boolean,
            default: false,
        },
        clicks_48h: {
            type: Number,
            default: 0,
        },
        clicks_1w: {
            type: Number,
            default: 0,
        },
        impressions_48h: {
            type: Number,
            default: 0,
        },
        impressions_1w: {
            type: Number,
            default: 0,
        },
        check_period: {
            type: Number,
            default: 24,
        },
        logs: [
            {
                date: { type: Date },
                reason: { type: String },
            },
        ],
    },
    cpmFloor: {
        type: Boolean,
        default: false,
    },
    lastFinalizedDate: { type: Date, index: true, default: Date.now },
},

mauticSynced: { type: Boolean, default: false },
mauticId: { type: String, default: "" },
epomId: { type: Boolean },
isEpomAdvertiser: {
    type: Boolean,
    default: false,
},
enableEpomVideoAd: {
    type: Boolean,
    default: false,
},

```

```

        reliablePublisher: {
            type: Boolean,
            default: false,
        },
        acceptNewTerms: {
            type: Boolean,
            default: false,
        },
        tokens: { type: Array, default: [] },
        pipedrivePersonId: {
            type: Number,
        },
        pipedriveSynced: {
            type: Boolean,
            default: false,
        },
        leadUser: {
            type: Boolean,
            default: false,
        },
    },
    { collection: "users" }
);

// create the model for users and expose it to our app
module.exports.User = mongoose.model("User", User);

const config = require("config");
const mongoose = require("mongoose");

class MongoConnection {
    constructor() {
        this.mongooseConnection = null;
    }

    async connect(log) {
        if (this.mongooseConnection) {
            return null;
        }
        try {
            this.mongooseConnection = await
mongoose.connect(config.get("mongoose.connectionUrl"),
config.get("mongoose.options"));
            this.mongooseConnection.connection.on("error", (error) =>
{
                log.error(`Mongo error: ${error}`);
            });
            this.mongooseConnection.connection.on("disconnected", ()
=> {
                log.info("Mongo disconnected.");
            });
            log.info(`Mongo successfully connected.`);
        } catch (err) {
            log.fatal(`Mongo connection error ${err}`);
            process.exit(1);
        }
        return true;
    }

    getConnection() {
        return this.mongooseConnection.connection;
    }
}

```

```
    async destroy() {  
      await this.mongooseConnection.connection.close();  
      await this.mongooseConnection.connection.removeAllListeners();  
      this.mongooseConnection = null;  
    }  
  }  
  module.exports.MongoConnection = MongoConnection;
```

Додаток 3
Копія презентації

НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
“КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ ІМЕНІ ІГОРЯ
СІКОРСЬКОГО”



ФАКУЛЬТЕТ ПРИКЛАДНОЇ МАТЕМАТИКИ

КАФЕДРА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ КОМП'ЮТЕРНИХ СИСТЕМ

Програмний метод формування рекомендацій реklamних оголошень на основі аналізу вподобань користувача

Доповідач: Дяченко Дмитро Олександрович

Науковий керівник: к.т.н., старший викладач, Н.А. Рибачок

Київ – 2020

АКТУАЛЬНІСТЬ ДОСЛІДЖЕННЯ

Наразі кількість інформації в мережі настільки велика, що людині дуже важко отримати те, що її дійсно цікавить. У зв'язку з цим, актуальною є розробка спеціальної рекомендаційної системи, що пропонуватиме рекламні оголошення, які відповідатимуть інтересам користувача. Такі персональні рекомендації формуються на основі аналізу вподобань поточного користувача та інших користувачів.

Об'єкт дослідження: процес формування особистих рекламних рекомендацій на основі вподобань користувачів.

Предмет дослідження: методи формування рекомендацій.



Наукове завдання: розробити програмний метод формування рекомендацій, який буде враховувати вподобання користувача стосовно рекламних товарів або послуг.

Мета дослідження: підвищення ефективності роботи рекламної системи за рахунок розробки програмного методу формування рекомендацій на основі аналізу вподобань користувача.

Окремі завдання

1. Виконати огляд існуючих методів роботи рекомендаційних систем, здійснити їх порівняльний аналіз та формалізувати задачу формування рекомендацій.
2. Розробити програмний метод побудови рекомендацій рекламних оголошень на основі аналізу вподобань користувача.
3. Провести порівняння ефективності роботи різних методів побудови рекомендацій.

НАЙБІЛЬШ ВІДОМІ РІШЕННЯ


























- Неперсоналізована система, загальна статистика
- Фільтрація на основі вмісту
- Колаборативна фільтрація
- Демографічна фільтрація

КЛАСИФІКАЦІЯ РЕКОМЕНДАЦІЙНИХ СИСТЕМ



КОЛАБОРАТИВНА ФІЛЬТРАЦІЯ

- використовує відомі оцінки групи користувачів для прогнозування невідомих уподобань іншого користувача
- основне припущення: ті, хто однаково оцінювали будь-які предмети в минулому, схильні давати схожі оцінки інших предметів і в майбутньому

ДЕМОГРАФІЧНА ФІЛЬТРАЦІЯ

Користувачі діляться на демографічні класи з точки зору їх особистих атрибутів.

Ці класи служать вхідними даними для процесу рекомендацій.

Мета процесу – знайти класи людей, яким подобається певний продукт.

Переваги:

- Немає потреби у наявності складних даних, таких як історія вподобань користувача або характеристики елементів системи

Недоліки:

- Загальна класифікація користувачів, втрачається індивідуальність

ФІЛЬТРАЦІЯ НА ОСНОВІ ВМІСТУ

Фільтрація на основі вмісту (CBF) використовує опис елементів, які раніше були переглянуті або придбані клієнтом і оцінені їм позитивно.

Система рекомендує споживачам товари, схожі на ті, які їм подобалися в минулому.

Переваги:

- Немає потреби у наявності даних про користувача, таких як вік, стать, а також не потребує даних про інших користувачів

Недоліки:

- Надспеціалізація

ГІБРИДНІ МЕТОДИ

Гібридний метод формування рекомендацій поєднує в собі декілька інших методів.

Основна ідея гібридних методів полягає в тому, щоб уникнути вад взятих за основу методів.

Існує безліч різних способів комбінування колаборативної, демографічної фільтрації та фільтрації на основі вмісту. Найбільш ефективними вважаються гібридні методи.

Пропонується розробити гібридний метод на основі колаборативної, демографічної фільтрації та фільтрації на основі вмісту.

МОДИФІКОВАНИЙ МЕТОД ФОРМУВАННЯ РЕКОМЕНДАЦІЙ



Фінальний прогнозований рейтинг оголошення обчислюється за формулою:

$$r_{us} = \frac{\alpha r_{us}^{DF} + \beta r_{us}^{CB} + \gamma r_{us}^{CF}}{\alpha + \beta + \gamma},$$

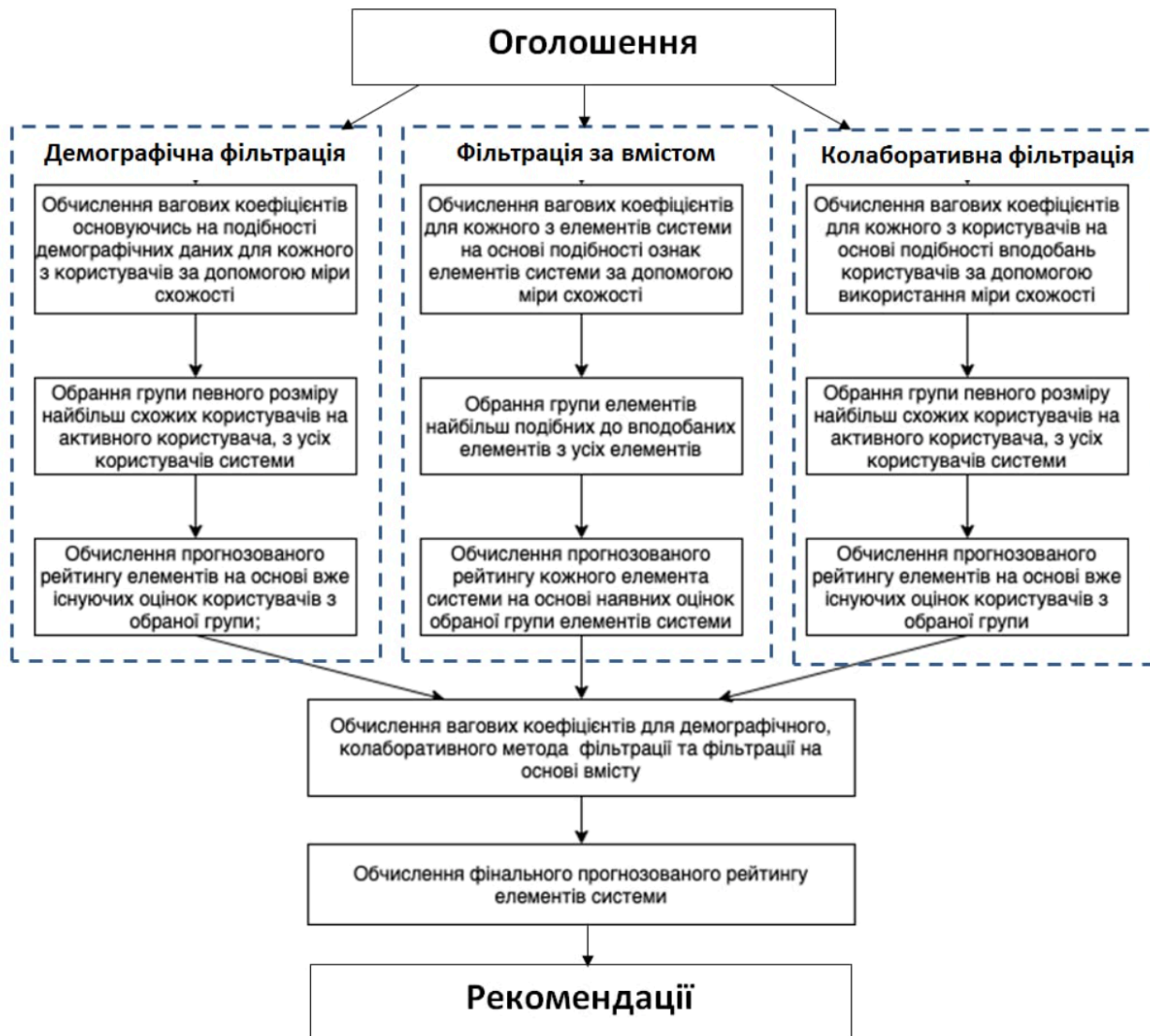
де r_{us}^{DF} – прогнозований рейтинг, отриманий за допомогою метода демографічної фільтрації,

r_{us}^{CD} – прогнозований рейтинг, отриманий за допомогою метода колаборативної фільтрації,

r_{us}^{CF} – прогнозований рейтинг, отриманий за допомогою метода фільтрації на основі вмісту,

α, β, γ – вагові коефіцієнти для кожного з методів відповідно.

СХЕМА РОБОТИ МОДИФІКОВАНОГО МЕТОДУ ФОРМУВАННЯ РЕКОМЕНДАЦІЙ



ОЦІНКА ЕФЕКТИВНОСТІ РОЗРОБЛЕНОГО МЕТОДУ. ДАНІ



Набір даних:

- 10 000 рекламних оголошень
- статистичні дані 50 000 користувачів

Використано:

- демографічні дані користувача
- дані про рекламні оголошення
- статистичні дані про вподобання користувачів:
 - ідентифікатор рекламного оголошення;
 - ідентифікатор користувача (fingerprint);
 - оцінка рекламного оголошення користувачем.

ОЦІНКА ЕФЕКТИВНОСТІ РОЗРОБЛЕНОГО МЕТОДУ. КРИТЕРІЇ



Для оцінки ефективності роботи методів будемо використовувати такі характеристики релевантності:

- точність : $P = \frac{t_p}{t_p + fp}$;
- повнота: $R = \frac{t_p}{t_p + fn}$;
- F-міра: $F = 2 * \frac{P * R}{P + R}$;

t_p – істинно-позитивні спрацювання,

f_p – хибно-позитивні спрацювання,

f_n – хибно-негативні спрацювання.

Ефективність будемо оцінювати за сумою цих трьох параметрів. Чим більший кожен з них, тим більшою вважається ефективність методу.

- ефективність: $E = P + R + F$

ОЦІНКА ЕФЕКТИВНОСТІ РОЗРОБЛЕНОГО МЕТОДУ. МЕТОДИ



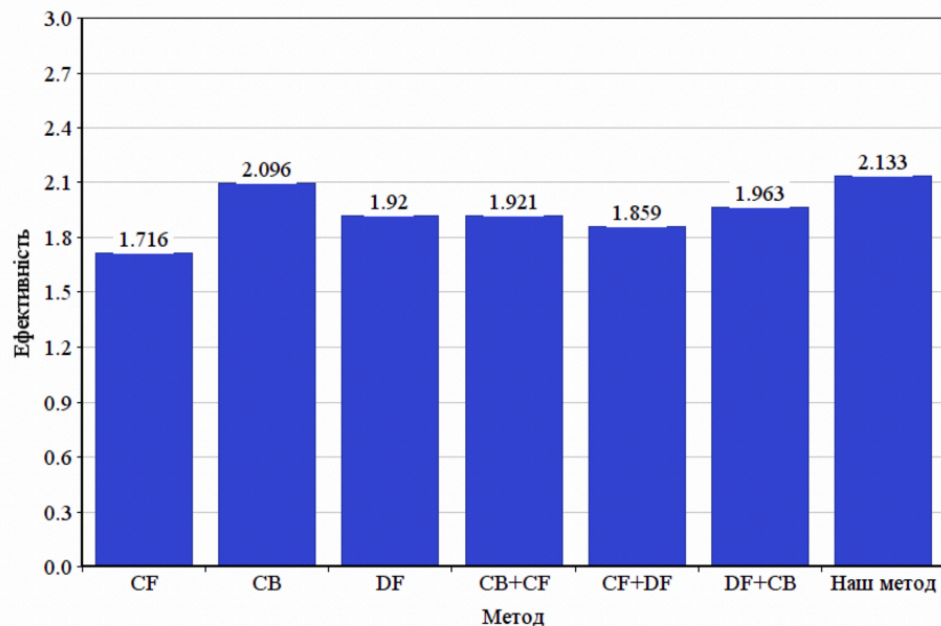
Методи застосовані для тестування:

- колаборативна фільтрація (CF);
- фільтрація на основі вмісту (CB);
- демографічна фільтрація (DF);
- CB+CF;
- CF+DF;
- DF+CB;
- розроблений модифікований метод (CF+CB+DF).

ОЦІНКА ЕФЕКТИВНОСТІ РОЗРОБЛЕНОГО МЕТОДУ. РЕЗУЛЬТАТИ 1



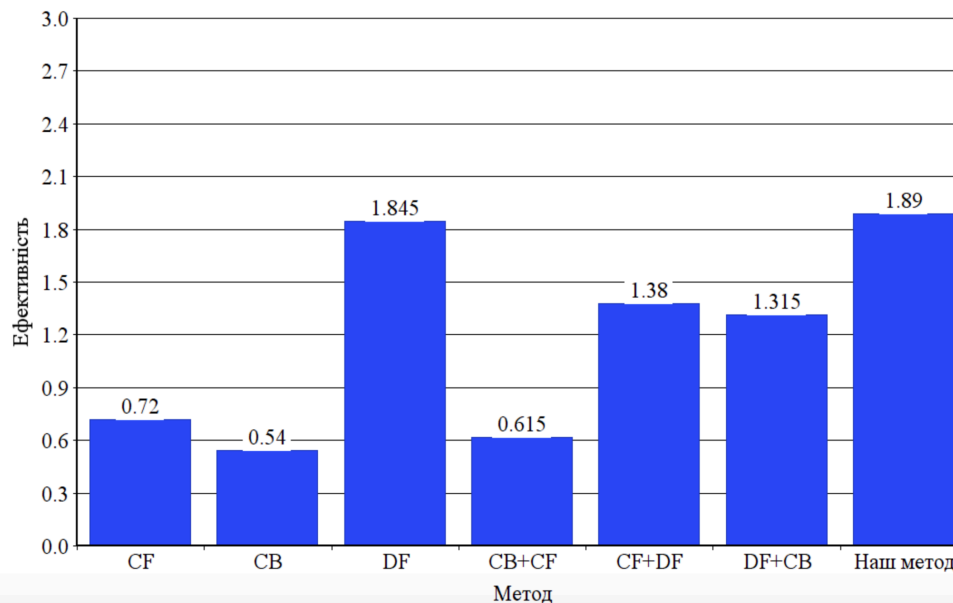
Алгоритм	Точність	Повнота	F-міра	Ефективність
CF	0.611	0.53	0.575	1.716
CB	0.712	0.69	0.794	2.096
DF	0.65	0.63	0.64	1.92
CB+CF	0.637	0.643	0.641	1.921
CF+DF	0.601	0.607	0.651	1.859
DF+CB	0.65	0.658	0.655	1.963
Модифікований алгоритм	0.728	0.697	0.708	2.133



ОЦІНКА ЕФЕКТИВНОСТІ РОЗРОБЛЕНОГО МЕТОДУ. РЕЗУЛЬТАТИ 2



Алгоритм	Точність	Повнота	F-міра	Ефективність
CF	0.23	0.25	0.24	0.72
CB	0.19	0.17	0.18	0.54
DF	0.61	0.62	0.615	1.845
CB+CF	0.2	0.21	0.205	0.615
CF+DF	0.47	0.45	0.46	1.38
DF+CB	0.45	0.46	0.455	1.315
Модифікований алгоритм	0.64	0.62	0.63	1.89



НАУКОВА НОВИЗНА



Наукова новизна одержаних результатів полягає у застосуванні гібридного методу до фільтрації даних для формування персональних рекомендацій рекламних оголошень. Розроблено метод, який поєднує в собі три методи фільтрації: колаборативну, демографічну та на основі вмісту.

АПРОБАЦІЯ РОБОТИ



Основні положення і результати роботи були представлені та обговорювались на XII науковій конференції магістрантів та аспірантів «Прикладна математика та комп'ютинг» ПМК-2020 (Київ, 18-20 листопада 2020 р.)

ВИСНОВКИ

Запропоновано метод формування рекомендацій рекламних оголошень на основі аналізу вподобань користувача.

Представлений метод є гібридним, тобто містить в собі агрегацію декількох методів побудови рекомендацій: колаборативного, демографічного методу фільтрації та фільтрації на основі вмісту.

Експериментально підтверджено переваги запропонованого методу над існуючими.

Практичне застосування результатів даного дослідження дозволить:

- користувачам отримувати персоналізовані рекомендації рекламних оголошень;
- рекламним платформам підвищити ефективність своїх рекламних оголошень.



СПИСОК ЛІТЕРАТУРНИХ ДЖЕРЕЛ

1. Гитис Л.Х. Кластерный анализ в задачах классификации, оптимизации и прогнозирования / Л.Х. Гитис. –М.: МГГУ, 2001. –104 с.
2. Дюран Б. Кластерный анализ. – М., 1977. – 128 с.
3. Jain, Murty, Flynn Data clustering: a review. ACM Computing surveys 31(3), 1999.
4. Журавлев Ю. И., Рязанов В. В., Сенько О. В. «Распознавание». Математические методы. Программная система. Практические применения. — М.: Фазис, 2006.



Дякую за увагу!